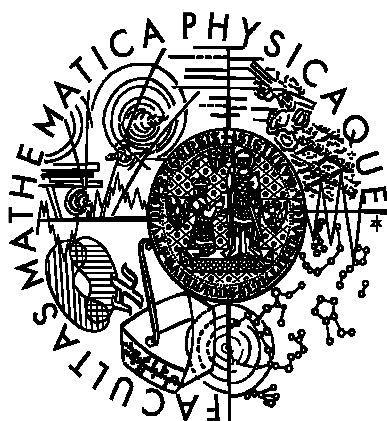


Univerzita Karlova v Praze  
Matematicko-fyzikální fakulta

## **BAKALÁŘSKÁ PRÁCE**



Peter Ďurik

### **Server pro online soutěže v programování**

Katedra aplikované matematiky

Vedoucí bakalářské práce: Mgr. Jakub Černý, KAM  
Studijní program: Informatika, Správa počítačových systémů

2007

Chcel by som poďakovať vedúcemu a konzultantovi bakalárskej práce Jakubovi Černému za podnetné rady pri realizácii projektu. Moje poďakovanie tiež patrí Martinovi Marešovi, ktorý je autorom časti aplikácie s názvom MO Eval.

Prehlasujem, že som svoju bakalársku prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím s požičiavaním práce a jej zverejňovaním.

V Prahe dňa 14.12.2007

Peter Ďurik

# OBSAH

<b>1</b>	<b>ÚVOD .....</b>	<b>4</b>
<b>1.1</b>	<b>Motivácia.....</b>	<b>4</b>
<b>1.2</b>	<b>Cieľ .....</b>	<b>5</b>
<b>2</b>	<b>VYHODNOCOVACÍ SERVER (ONLINE JUDGE) .....</b>	<b>6</b>
<b>2.1</b>	<b>Ako funguje vyhodnocovací server .....</b>	<b>6</b>
<b>2.2</b>	<b>Existujúce servery.....</b>	<b>8</b>
<b>3</b>	<b>OCPS.....</b>	<b>10</b>
<b>3.1</b>	<b>Komponenty aplikácie .....</b>	<b>10</b>
<b>3.2</b>	<b>OCPS webový server .....</b>	<b>11</b>
3.2.1	Administrácia.....	11
3.2.2	Prostredie pre riešiteľov .....	14
3.2.3	Implementácia.....	15
<b>3.3</b>	<b>OCPS Manager.....</b>	<b>17</b>
3.3.1	Cyklus OCPS Managera .....	18
<b>3.4</b>	<b>Evaluátor.....</b>	<b>20</b>
<b>4</b>	<b>UŽÍVATEĽSKÁ DOKUMENTÁCIA.....</b>	<b>21</b>
<b>4.1</b>	<b>Systémové požiadavky .....</b>	<b>21</b>
4.1.1	Server .....	21
4.1.2	Klient.....	21
<b>4.2</b>	<b>Inštalácia .....</b>	<b>21</b>
4.2.1	Webový server.....	21
<b>4.3</b>	<b>Konfigurácia .....</b>	<b>30</b>
4.3.1	Konfigurácia OCPS webového serveru .....	30
4.3.2	Administrácia obsahu.....	32
4.3.3	OCPS manager .....	42
4.3.4	Vyhodnocovací server OCPS (JUDGE) .....	43
<b>4.4</b>	<b>Práca s aplikáciou .....</b>	<b>44</b>
4.4.1	Užívateľské webové rozhranie. ....	44
4.4.2	OCPS manager .....	51
<b>5</b>	<b>ZÁVER.....</b>	<b>53</b>
<b>6</b>	<b>ZDROJE .....</b>	<b>55</b>

**Názov práce:** Server pro online soutěže v programování

**Autor:** Peter Ďurík

**Katedra:** Katedra Aplikované Matematiky

**Vedúci bakalárskej práce:** Mgr. Jakub Černý

**E-mail vedúceho:** Jakub.Cerny@mff.cuni.cz

**Abstrakt:** Cieľom práce je navrhnúť a zostaviť aplikáciu pre vyhodnocovanie a správu programátorských súťaží. Aplikácia by mala byť navrhnutá tak, aby bola rovnako dobre použiteľná na programátorských súťažiach ale aj na vyhodnocovanie domácich úloh z cvičení z programovania. Po preskúmaní existujúcich riešení je potreba navrhnúť systém, ktorý bude obsahovať príjemné užívateľské prostredie pre správu aplikácie a prostredie pre súťažiacich. Systém by mal vyhodnocovať riešenia dostatočne spravodlivo a vyhodnocovanie by malo byť odolné voči podvodným riešeniam.

**Kľúčové slová:** online evaluátor, online judge, programátorská súťaž, vyhodnocovanie riešení

**Title:** Server for online competitions in programming

**Author:** Peter Ďurík

**Department:** Department of Applied Mathematics

**Supervisor:** Mgr. Jakub Černý

**Supervisor's e-mail address:** Jakub.Cerny@mff.cuni.cz

**Abstract:** The aim of this project is to design and to create an application for evaluating and administration of programming competitions. Application should be designed to be used in programming contests and for evaluating homework from lessons of programming as well. After look into existing solutions it is needed to design system that will contain a comfortable user interface for administration of application and interface for contestants. The system should evaluate solutions enough rightly and evaluating should be resistant against fake solutions.

**Keywords:** online evaluator, inline judge, programming competition, evaluation of solutions

# 1 ÚVOD

## 1.1 Motivácia

Z vlastnej skúsenosti viem, aké je zložité učiť sa niečo nové. Ako človek zameraný na zmyslové vnemy potrebujem príklad, potrebujem si svoje vedomosti overiť. To sa týka hlavne praktickej činnosti. Programovanie ako praktická činnosť do určitej miery je. Zaiste je potreba teoretické znalosti, ale v konečnom dôsledku ide o vytvorenie i keď nie fyzického ale predsa diela.

Učiť sa programovať znamená naučiť sa programovací jazyk ale hlavne naučiť sa myslieť „strojovo“. Kto touto cestou prešiel, určite si spomenie aké obtiažne bolo pochopiť, čo sa za tými riadkami kódu skrýva. V podstate je potrebné sa naučiť myslieť ako stroj. No a overiť si svoje nadobudnuté znalosti je potreba určite prakticky. To platí vždy a všade. Teória proste nestačí a je teda užitočné vyriešiť nejaké praktické príklady. Koľkokrát sú ale k dispozícii len zadania. V tomto prípade je komplikované sa dozvedieť, či máme úlohu vyriešenú správne alebo nie. Chce to ešte niečo. Buď budeme mať k dispozícii vzorovo vyriešenú úlohu alebo vzorové vstupy a k nim odpovedajúce výstupy. V prvom prípade jednoducho skontrolujeme náš výsledok s výsledkom vzorového riešenia na rovnaké vstupné dáta. Mať vzorové riešenie navádza k tomu aby človek kopíroval. Iným spôsobom ako si skontrolovať správnosť riešenia je mať k dispozícii sadu vstupných dát a k nim sadu očakávaných výstupných dát. Potom už len stačí na zadané vstupné dáta porovnať naše výstupy s danými a výsledok je na svete.

Keď už programovať vieme a baví nás to, môžeme si zrovnať sily s inými. K tomu slúžia všemožné programátorské súťaže a semináre. K dispozícii je zoznam úloh, ktoré je potreba vyriešiť a riešenie zaslať organizátorom k vyhodnoteniu. V dnešnej dobe technológií sa nikto nebude diviť, keď poviem, že takéto súťaže majú často krát vo svojom slove označenie „On-line“. Čo ale to slovo s spojením so seminárom, či súťažou znamená? Môže to znamenať, že sa niekde zaregistrujeme, a po internete posielame riešenia, ktoré sú nejakým spôsobom vyhodnocované. Alebo to môže znamenať, že riešenia nevyhodnocujú manuálne ľudia, ale automatizované počítačové programy.

Rovnako tak cvičenia z programovania na školách. Žiaci dostávajú rôzne úlohy a ich úlohou je vyriešiť ich a zaslať učiteľovi k vyhodnoteniu. Nebolo by pekné

mať systém, ktorý urobí tú prácu čiastočne za vyučujúceho? Žiaci zašlú riešenie do systému a on ho vyhodnotí a zaznamená výsledky.

## **1.2 Cieľ**

Predchádzajúce úvahy ma viedli k tomu, aby som napísal túto prácu. Budem sa zaoberať automatickým vyhodnocovaním úloh z programovania. Cieľom práce je vyskúmať z čoho takýto systém pozostáva a pokúsiť sa zostaviť funkčnú verziu systému, ktorá by mala nasledujúce vlastnosti:

- úložisko rôznorodých problémov a úloh z programovania
- automatické vyhodnocovanie zaslaných riešení
- integrácia užívateľov do prostredia
- dostupnosť z ľubovoľného počítača pripojeného k Internetu
- príjemné užívateľské prostredie
- možnosť použitia pri výučbe programovania

Ide teda o vyhodnocovací server pre online súťaže v programovaní a rovnako tak o systém slúžiaci k výučbe alebo precvičovaniu si úloh z programovania. Aplikácia by mala z technickej stránky obsahovať klientskú, obslužnú časť, ktorá sa bude deliť na užívateľskú a potom serverovú, výkonnú časť, slúžiacu k vyhodnocovaniu úloh. Z užívateľského pohľadu správcu by mala podporovať správu užívateľských účtov, zadaní problémov či úloh k riešeniu. Z pohľadu riešiteľa by mala obsahovať prehľad úloh k riešeniu, zasielanie riešení k vyhodnoteniu či zobrazovanie výsledkov. Takto navrhnutý systém by bol dobre použiteľný v školách pri výučbe programovania alebo tiež ako server slúžiaci k rôznym programátorským seminárom alebo súťažiam.

## **2 VYHODNOCOVACÍ SERVER (ONLINE JUDGE)**

### **2.1 Ako funguje vyhodnocovací server**

Vyhodnocovací server ako kompletná aplikácia pozostáva z viacerých častí, ktoré sú navzájom medzi sebou prepojené.

Nenahraditeľnou časťou je užívateľské rozhranie. Cez to riešiteľ získava zadania úloh, zasiela vyriešené úlohy k vyhodnoteniu, prezerá si výsledky. Toto užívateľské rozhranie je tvorené aplikáciou na webovom serveri. Užívateľ potom k nemu pristupuje pomocou internetového prehliadača. V podstate pracuje s webovou stránkou.

Užívateľské rozhranie je napojené na nejakú databázu, ktorá slúži ako zhromaždisko dát. Uchovávajú sa v nej všetky potrebné údaje, ktoré sa zobrazujú užívateľovi. Je to napríklad zoznam úloh k riešeniu či zoznam užívateľov a ich osobné údaje. Všetky zaslané úlohy k vyhodnoteniu od jednotlivých riešiteľov a ich výsledky. A v neposlednej rade rôzne nastavenia samotnej aplikácie.

Z nášho pohľadu najdôležitejšou časťou je samotný vyhodnocovací mechanizmus - evaluátor. Je často nazývaný aj Judge. Ten je srdcom celej aplikácie a ako už jeho názov napovedá, slúži k samotnému vyhodnocovaniu vyriešených úloh.

Ako vlastne funguje také vyhodnocovanie? Riadiaca časť aplikácie predá evaluátoru zdrojový kód úlohy, ktorý sa bude vyhodnocovať. Teda naprogramované riešenie úlohy, čo zaslal riešiteľ cez užívateľské rozhranie. Evaluátor obdrží zdrojový kód úlohy spolu s doplnkovými informáciami. Musí vedieť o riešení akého konkrétneho problému sa jedná, aby ho mohol patrične vyhodnotiť. Ďalej je potrebné, aby evaluátor vedel, aké parametre bude samotné vyhodnocovanie mať. Konkrétne sa jedná o programovací jazyk, v akom bola úloha naprogramovaná. Jedná sa predovšetkým o kompilované programovacie jazyky ako C, Java či Páscal. Medzi ďalšie potrebné údaje patrí počet testov, ktoré sa prevedú s daným riešením a vstupné i výstupné dáta, ktoré budú použité pri testoch.

Zaslaná úloha sa teda podľa typu programovacieho jazyka skompiluje. Pri kompilácii sa použijú parametre, ktoré sú k danej úlohe prislúchajúce. V tejto fáze už teoreticky proces vyhodnocovania môže skončiť. A to v prípade, že sa nepodarí úlohu skompilovať. Pokiaľ kompilácia dopadne úspešne, nastáva séria testov na rôznorodé vstupné dáta.

Skompilovaná vyriešená úloha sa spustí s prislúchajúcimi vstupnými dátami a výstupy z každého testu sú uchované. Tie sa potom porovnajú s očakávanými výstupmi a v prípade zhody je úloha označená ako správne vyriešená. Ak sa výstupy nezhodujú, úloha je označená ako vyriešená nesprávne. To však nie je kompletný stav ako môže byť hodnotená úloha označená.

Počas spustenia skompilovanej úlohy môže dôjsť k rôznym stavom. Pokiaľ je úloha zle naprogramovaná a nejakým spôsobom sa zacyklí, alebo je algoritmus tak komplikovaný že v určitom časovom limite nie sú k dispozícii výstupné dáta. To znamená, že testovaný program sa ešte neukončil. Beh tohto programu je ukončený automaticky a úloha je označená stavom o vypršaní časového limitu. Tento údaj sa predá evaluátoru ako parameter a môže sa pri každej úlohe líšiť. Dokonca sa môže líši v každom teste, ak sa úloha testuje sériou testov.

Ďalším problémom, ktorý môže nastať je, že program bude alokovať neúmerne množstvo pamäti. To sa ľahko môže stať. Preto ďalší parameter udáva, koľko maximálne operačnej pamäti môže úloha alokovať. Pokiaľ bude počas testu dosiahnuté toto číslo, bude beh programu prerušený a úloha je označená stavom o naplnení limitu alokácie operačnej pamäti.

Pri testovaní sa ako obmedzenie okrem času behu programu a veľkosti operačnej pamäti stanovuje diskový priestor v ktorom má testovaný program sa právo pohybovať. Nie je povolené, aby program všemožne komunikoval s okolím a je preto uzavretý len na obmedzený priestor. Pokiaľ bu sa teda program pokúsil prístupíť k nejakým súborom mimo obmedzený priestor, bude ukončený a jeho stav označený ako pokus pristupovať k zakázaným súborom.

No a posledná možnosť, ktorá môže nastať je, že program sa spustí a v zapätí skončí nejakou behovou chybou. Je v tom prípade označený stavom, že skončil behovou chybou.

Výsledok testu či série testov sa pošle naspäť riadiacej časti programu a je zaznamenaný, uschovaný v databáze a riešiteľ sa môže naň pozrieť.



## 2.2 Existujúce servery

Keď som pátral po existencii takýchto systémov, zistil som, že ich už niekoľko existuje. Sú zamerané ako súťažné servery. Ide o systémy udržiavané prevažne univerzitami. Ako príklady môžem uviesť nasledujúce:

- UVa Online Judge [1]  
Valaloid judge, jeden s najstarších systémov
- El Judge [2]  
Online MIPT programming contest
- FZU Online Judge [3]  
Fuzhou University
- ZHEJIANG niversity Online Judge [4]  
*<http://acm.zju.edu.cn>*
- TJU Online Judge [5]  
Tiabjin University
- Timus Online Judge [6]  
Ural State University
- Sphere Online Judge [7]
- Online Contester [8]  
Satarov State University

Všetky fungujú na rovnakom princípe, ako je popísane v [9]. Po prihlásení sa zaregistrovaného užívateľa do aplikácie je k dispozícii sada problémov rôznych obtiažností. Je len na riešiteľovi, ktorý si vyberie a pokúsi sa ho vyriešiť. Následne zašle riešenie a systém ho vyhodnotí. Výsledok môže byť rovnaký ako je popisované v predošlej kapitole. Každá úloha obsahuje štatistiku, kde je percentuálne a v počtoch zobrazené, koľko pokusov skončilo s akým výsledkom. Navyše tieto servery umožňujú vybrať si v akom jazyku sa bude daná úloha riešiť. Je možné si vybrať z niekoľkých možností, typicky ANSI C, C++, JAVA či PASCAL. Rovnako táto informácia je zahrnutá v štatistikách pri danej úlohe, koľko riešení bolo zaslané v akom jazyku. Jedná sa len o štatistiky vyhodnotených riešení v čom sa jednotlivé servery líšia. Závisí to tak len na implementácii, čo všetko sa rozhodnú prevádzkovatelia skúmať a zobrazovať. No a rozdiely sú i v počte podporovaných programovacích jazykov. Napríklad

*Sphere Online Judge* ich podporuje viac ako 30. Inak sa dajú tieto systémy považovať za totožné.

Ako výhoda týchto serverov sa javí skutočnosť, že skoro v úplnej anonymite je možné si pre zábavu vyriešiť nejakú programátorskú úlohu. V zozname úloh sa dajú nájsť úlohy od tých najľahších, ktoré zvládnu i začiatočníci až po tie najzložitejšie, ktoré si vyžadujú bystrý úsudok a určitú dávku skúseností v programovaní.

A nevýhody? Nevýhody snáď ani v pravom slova zmysle neexistujú. Tak ako sú systémy postavené a účel, ktorý majú plniť, plnia si myslím svedomite. Problém je, že ja hľadám riešenie, ktoré sa bude dať aplikovať rovnako tak do škôl na vyučovanie, ako i na súťaže či semináre.

Taký systém je vyvíjaný Na Univerzite Karlovej na Matematicko-fyzikálnej fakulte a jeho názov je CodEx. Tento projekt zastrešuje KSVI a slúži ako systém pre vyhodnocovanie úloh z KSP (Korespondenční Seminář z Programování) a rovnako tak ho bude možné používať na vyhodnocovanie úloh z cvičení z programovania. Projekt bol vyvíjaný v čase, keď vznikala moja práca a v počiatku som si z neho vzal niekoľko myšlienok ako inšpiráciu.

Ďalej som sa už pustil vlastnou cestou a premýšľal, ako taká aplikácia môže vypadáť a čo všetko by mala obsahovať.

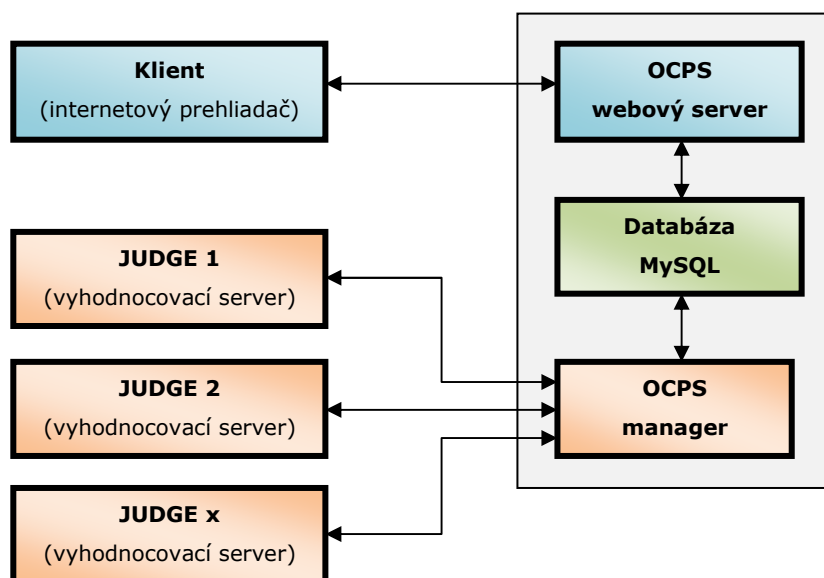
## 3 OCPS

OCPS (Online Contest Programming Server), neskôr pomenovaný ako OCtoPuS, je aplikácia webová. Je vyvíjaná za účelom použitia pri cvičeniach v programovaní alebo ako aplikácia slúžiaca pre verejné súťaže, či semináre z programovania. Pri návrhu bolo potrebné rozobrať fakty, čo všetko bude musieť aplikácia vykonávať a podľa toho ju rozdeliť na jednotlivé časti, či celky, ktoré budú fungovať ako logicky samostatné komponenty. Z viacerých dôvodov je k fungovaniu aplikácie potrebný operačný systém Linux.

### 3.1 Komponenty aplikácie

OCtoPuS sa skladá z viacerých častí, ako je znázornené na diagrame 3-1. Jedna časť sídli na webovom serveri, tá komunikuje s užívateľom. Ďalšia časť je vyhodnocovací server *JUDGE*, ktorý prevádza samotné vyhodnocovanie riešení. Ďalej je to *OCPS manager*, ktorý zabezpečuje zber požiadaviek na vyhodnocovanie a komunikuje s vyhodnocovacími servermi. Všetko je prepojené databázou MySQL.

*OCPS webový server*, *databáza MySQL* a *OCPS manager* musia byť fyzicky umiestnené na jednom stroji. Naopak každý *Vyhodnocovací server JUDGE*, ktorých môže byť súčasne viac, musí bežať na samostatnom stroji.



3-1: schéma komponent aplikácie

## 3.2 OCPS webový server

Táto časť aplikácie, ako už názov napovedá, je umiestnená na webovom serveri. OCPS webový server integruje prostredie pre komunikáciu s riešiteľom a tiež administratívne rozhranie pre správu aplikácie. Užívateľ k nemu prístupuje pomocou webového prehliadača.

### 3.2.1 Administrácia

Administratívne rozhranie umožňuje aplikácii spravovať jednotlivé entity. Po prihlásení sa do aplikácie je možné si menu vyberať čo sa bude spravovať.

#### Užívatelia

Aplikácia musí rozlišovať užívateľa podľa toho či sa jedná o administrátora, o takzvaného super užívateľa (v reálnom používaní to bude napríklad učiteľ) alebo o riešiteľa. Z toho dôvodu je každý užívateľ v databáze označený príznakom *admin* ktorý je nastavený na hodnotu *nula*, ak sa jedná o riešiteľa alebo *jedna*, ak ide o administrátora alebo super užívateľa. Na základe toho sa po prihlásení do aplikácie cez vstupnú prihlasovaciu stránku dostane buď do administratívnej sekcie alebo do sekcie pre riešiteľov.

Ako hlavný administrátor je vytvorený užívateľ *root*, podobne ako je to v operačnom systéme Linux. Ten má maximálne práva pri spravovaní aplikácie. Jediné čo by nemal byť schopný vykonať je odobrať sám sebe práva administrátora a zmazať svoj účet.

Každý vytvorený užívateľ je jednoznačne identifikovaný svojím prihlasovacím menom a heslom. Ďalej sa v aplikácii uchováva jeho celé meno a emailová adresa. Tieto údaje sa dajú v administratívnej časti aplikácie meniť. To môžu robiť užívatelia s administrátorskými privilégiami. Nemôžu však upravovať iných administrátorov ani ich mazať. A nemajú právo nastavovať užívateľom administrátorské privilégia. Rovnako nemajú právo odobrať administratívne privilégia sami sebe či zmazať svoj vlastný účet. To môže len užívateľ *root*.

#### Skupiny

Ak sa má systém dať používať na cvičeniach z programovania, je nutné zostaviť skupiny užívateľov do ktorých bude riešiteľ patriť. To neplatí pre užívateľov s administrátorskými právami. Oni nemajú príslušnosť k žiadnej skupine. Entita

skupina teda obsahuje zoznam riešiteľov, ktorý do nej patria, pričom riešiteľ môže v jednom okamihu patriť i do viac ako jednej skupiny. Pre účely online seminárov je možné vytvoriť zvláštnu skupinu a tak je systém použiteľný i v tomto smere. Administrátor môže skupinu ľubovoľne editovať, pričom do nej môže pridávať či z nej odoberať riešiteľov.

## **Problémy**

Ďalšiu entitu tvoria takzvané problémy. K nim majú prístup užívatelia s administrátorskými právami. Problém je inštancia úlohy bez príslušnosti ku skupine. Takže je pre riešiteľov neviditeľná. Problém obsahuje zadanie, popis vstupov a výstupov a tiež vzorový vstup a k nemu prislúchajúci výstup.

## **Kategórie problémov**

Problémy je možné umiestňovať do kategórií. Kategórie tvoria stromovú štruktúru a na konci každej vetvy v štruktúre je možné umiestniť problémy. Ak bude problémov príliš mnoho, bude obtiažne sa v nich orientovať. Preto je dobré ich priradovať do kategórií. Upravovanie kategórií má svoje pravidlá. Pokiaľ bude potrebné nejakú kategóriu premiestniť v strome inam, je to možné len pokiaľ cieľová kategória neobsahuje úlohy. Teda pokiaľ nie je listom. Rovnako tak mazanie musí byť možné, len ak je kategória na konci hierarchie a neobsahuje momentálne žiadne problémy.

## **Úlohy**

Aby riešiteľ v danej skupine mohol niečo riešiť, je potrebné vytvoriť úlohu. Úlohy sú vlastne problémy (zmieňované vyššie), ktoré sú priradené k určitej skupine.

Každá úloha obsahuje odkaz na problém z ktorého je vytvorená a navyše obsahuje údaje o začiatku a konci. Jedná sa o údaje, ktoré hovoria o tom, či je úloha aktívna. Toto nachádza svoje uplatnenie napríklad pri cvičeniach z programovania, kedy sa určí, od akého času a dátumu bude možné zasielať riešenia úlohy a do kedy je to možné. Rovnako to dáva možnosť vytvoriť úlohy pre skupinu do budúcnosti, nastavením času začiatku. Tak je možné pripraviť si pre žiakov úlohy dopredu bez toho, aby mohli čo i len vidieť ich zadania.

Úloha ďalej obsahuje príponu súboru. Jedná sa o akési explicitné učenie programovacieho jazyka, v ktorom sa bude úloha riešiť. Tento údaj slúži viac menej evaluátoru aby systém automaticky nastavil príponu zasielanej úlohy

bez ohľadu na to, ako ju nastaví riešiteľ. Tu nachádzame rozdiel medzi touto aplikáciou a zmieňovanými *online judge* systémami, kde je možné úlohu riešiť v nami zvolenom programovacím jazyku. V skutočnosti to ale nie je tak úplne pravda. Musíme si uvedomiť, že v *online judge* systémoch neexistuje pojem *úloha*. Sú tam len problémy. Kdežto tu je úloha ako inštancia problému s parametrami navyše. Aby bolo možné rovnaký *problém* riešiť i vo viac programovacích jazykoch, je potrebné vytvoriť dve rôzne úlohy z toho istého problému a priradiť im rôzne jazyky. V skutočnosti, ak už je nejaká úloha vytvorená pre skupinu cvičenia z programovania, typicky je to cvičenie pre jeden programovací jazyk. Nemá teda význam dávať možnosť zaslať riešenie i v inom jazyku. Ďalším dôvodom príslušnosti úlohy len k jednému programovaciemu jazyku je samotný evaluátor. Ku každej úlohe totiž existuje jeden konfiguračný súbor, v ktorom sa mimo iné uvádza kompilátor.

Jediné obmedzenie tak nastáva pre online súťaže či verejné semináre v programovaní. Tu sa naskytá riešenie vytvoriť pre každý programovací jazyk zvlášť skupinu a od každého problému vytvoriť úlohu do príslušnej skupiny pre podporované programovacie jazyky. Ani v tomto prípade o nič neprichádzame. Či už hovoríme o vytváraní štatistík, alebo prehľadnosti aplikácie. Jediné obmedzenie je vo vytváraní viac úloh. Ale to sa dá napraviť implementovaním vytvorenia takzvanej multi-úlohy, kde by sa v jednom kroku z problému vytvorilo súčasne viac úloh z jedného problému.

Aby nebolo určenie programovacieho jazyka len pomocou prípony súboru mäťúce, obsahuje každá úloha komentár, kde je možné to okrem iného špecifikovať.

No a posledným parametrom úlohy je maximálny počet zaslaní riešenia k vyhodnoteniu. Tu je možné špecifikovať koľko krát je možné úlohu zaslať. Toto nachádza uplatnenie v cvičeniach, ak chceme zamedziť bezcieľnému zahlcovaniu aplikácie nevychovanými riešiteľmi. Pokiaľ z akýchkoľvek dôvodov nechceme toto obmedzenie uplatniť nastavíme túto hodnotu na vysoké číslo, ktoré za normálnych okolností nie je možné dosiahnuť. Zasielanie úloh k vyhodnoteniu má ešte jedno obmedzenie. Jedná sa o vlastnosť aplikácie, kde nie je možné zaslať riešenie úlohy pokiaľ ešte nebola vyhodnotená predchádzajúca. To znamená že vo fronte požiadaviek sa nemôžu súčasne nachádzať dva záznamy s rovnakým identifikátorom užívateľa a úlohy.

## **Evaluátor**

Súčasťou administratívneho prostredia je i spáva evaluátorov. Evaluátor je samostatná časť aplikácie ako bolo spomínané, je ich možné zároveň používať viac naraz. Preto je v databáze uložený ich zoznam, ktorý obsahuje IP adresu počítača, na akom je evaluátor nainštalovaný, a užívateľa operačného systému na danom počítači, pod ktorým sa evaluátor spúšťa.

### **3.2.2 Prostredie pre riešiteľov**

Ak sa do aplikácie prihlási užívateľ bez administratívnych práv, je označený za riešiteľa. Automaticky je teda presmerovaný do prostredia pre riešiteľov.

To obsahuje v úvode výber medzi skupinami, kam riešiteľ patrí. Pre každú skupinu existuje jej popis.

Následne je možné sa dostať k úlohám. Keďže úlohy môžu byť v troch rôznych stavoch, sú i aplikácii tak zobrazované. Úloha môže byť neaktívna a to v zmysle, že ešte momentálny čas nedosiahol času aktivácie úlohy. Takéto úlohy nie sú zobrazované. Ďalej sa jedná o úlohy, ktoré sú aktívne a o úlohy už neaktívne, ktoré sú uzavreté a nie je ich možné vyhodnocovať. Obe posledne spomínané skupiny sú zobrazované v prehľadných tabuľkách, kde je uvedený čas začiatku a konca aktivity úlohy, symbol, či bola úloha úspešne vyriešená a odkazy na manipuláciu s úlohou.

### **Zobrazenie zadania**

Medzi akcie na manipuláciu s úlohou patrí zobrazenie jej zadania. Tu sa zobrazia informácie problému z ktorého je úloha vytvorená. Na základe týchto je riešiteľ schopný úlohu vyriešiť. Hneď na vrchu stránky sú odkazy pre rýchlu manipuláciu, ktoré umožňujú zaslať riešenie, či zobraziť výsledok posledného vyhodnocovania.

### **Zaslať riešenie úlohy**

Tu sa zobrazí prvok ktorý umožňuje zaslať súbor so zdrojovým kódom úlohy k vyhodnoteniu. Ale to len v prípade že nebola porušená niektorá podmienka. Je dôležité kontrolovať, či užívateľ chce zasielať úlohu, ktorá patrí do jeho skupiny, tiež je potrebné kontrolovať, či daná úloha od neho nebola zaslaná k vyhodnoteniu a momentálne čaká vo fronte. V takom prípade je riešiteľ

upozornený, že akcia nie možná. Rovnako tak sa môže snažiť zaslať úlohu, ktorá už nie je aktívna. Pretože to, s akou úlohou sa práve manipuluje, určuje jej ID, ktoré sa zadáva ako parameter URL. Môže tak nastať situácia, keď sa bude riešiteľ podvrhom manipulovať s úlohou, s ktorou to za normálnych okolností nie je možné.

Na stránke zo zoznamom úloh v tabuľke je tlačidlo pre zaslanie neaktívne, pokiaľ úloha už nie je aktívna, alebo už bola úspešne vyriešená.

Súbor so zdrojovým kódom zaslanej úlohy sa uloží do s presne vygenerovaným názvom a príponou, odpovedajúcou nastaveniu pre danú úlohu. Kde sa súbory uchovávajú a ako sa generuje názov, bude popísané neskôr v časti o OCPS Manager (Kapitola 3.3).

## **Vyhodnotenie úlohy**

Na tomto mieste sa zobrazí výsledok z posledne zaslanej úlohy k vyhodnoteniu. Je možné, že úloha ešte nie je vyhodnotená. V tom prípade sa zobrazí len informácia, že úloha čaká vo fronte. Rovnako sa môže stať, že úloha ešte vôbec nebola k vyhodnoteniu zaslaná a vo tom prípade sa táto informácia pri pokuse o získaní výsledku vyhodnocovania zobrazí.

V zázname o vyhodnotení úlohy sa nachádzajú informácie o čase zaslania úlohy k vyhodnoteniu, o čase vyhodnotenia úlohy, pridelené body a výsledok akým evaluátor ohodnotil úlohu. V prípade výsledku označujúceho chybu pri kompilácii je zobrazená informácia kompilátora o chybe.

## **Zoznam zaslaných riešení**

Toto je zoznam všetkých doteraz zaslaných riešení danej úlohy. Obsahuje časovo zostupne zoradený zoznam jednotlivých výsledkov tak, ako je to v prípade predchádzajúcom. Tam sa však jedná len po posledne zaslané riešenie.

### **3.2.3 Implementácia**

OCPS webový server je naprogramovaný v jazyku PHP, ktorý sa k danému účelu hodí z dôvodu dobrej integrácie do webového servera Apache. Jazyk je dostatočne známy a rozšírený, preto ktokoľvek môže aplikáciu poopraviť, ak to



bude potrebovať. Jazyk dobre komunikuje s databázou MySQL, ktorú som sa rozhodol použiť ako úložisko dát.

Každá stránka je ošetrená kontrolou oprávnenia prístupu. V administračnej časti je kontrola prihláseného užívateľa, či má administračné práva. Tie sa ukladajú ako príznak pri prihlásení sa do aplikácie v *session*. V *session* premenných je uložené i prihlasovacie meno a heslo. Meno a heslo sú šifrované resp. dešifrované privátnymi metódami `_encode` resp. `_decode`. Pre šifrovanie sa používa externý modul `Crypt_Blowfish`, ktorý pre svoj beh nevyžaduje PHP rozšírenie `mcrypt`. Pokiaľ je toto rozšírenie k dispozícii, bude použité.

Pre prácu so *session* je vytvorená trieda aby sa zamedzilo priamemu prístupu k hodnotám v nej uložených.

V prípade riešiteľa sa v *session* ukladá aktuálna skupina, s ktorou sa pracuje a aktuálne vybraná úloha. Pri prístupe k nejakému problému sa tak na každej stránke kontroluje či je manipulované s úlohou či skupinou, ku ktorej má riešiteľ oprávnenie, alebo či manipulovaná úloha patrí do danej skupiny v ktorej sa riešiteľ momentálne nachádza. Tak sa zamedzí neoprávnenej manipulácii s objektmi, kde nie sú práva, nakoľko sa k objektom pristupuje cez ich identifikátor v parametri URL a je tak možná potenciálna machinácia zo strany riešiteľa.

Takéto kontroly sú aplikované i v prípade administračnej časti. Ak je napríklad v sekcii pre mazanie užívateľa umelo zmenený identifikátor objektu označujúci napríklad na užívateľa aktuálne prihláseného, je zobrazené chybové hlásenie. Nie je možné aby užívateľ zmazal sám seba.

Po prihlásení sa do aplikácie sa do *session* uloží aktuálny čas. V konfiguračnom súbore sa nachádza čas po ktorom *session* vyprší. Pokiaľ je užívateľ po túto dobu neaktívny a následne prejde na inú stránku je automaticky odhlásený a presmerovaný na prihlasovaciu stránku. Pokiaľ je pri prechode na inú stránku *session* platná je nastavený aktuálny čas a tým začne doba expirácie plynúť odznova.

## Šablóny

Aplikácia je naprogramovaná tak, aby bola oddelená výkonná časť od prezentačnej. K tomu účelu je využitý systém šablón. Prezentované dáta sú

vyprodukované v PHP a ďalej sa predajú šablónovaciemu systému *ETS*. Ten vyprodukuje html stránku, ktorá je prezentovaná vo webovom prehliadači.

Tento systém bol vybraný:

- pretože je napísaný v PHP nie je potreba žiadnej ďalšej technológie na webovom serveri
- podporuje konštrukcie, ktoré sú známe z bežných programovacích jazykov
- podpora polí
- rôzne podmiennečné vyhodnocovanie (IF, WHILE, ...)
- formátovanie dát
- prístup do akýchkoľvek úrovni dátového stromu
- možnosť pracovať s rekurzívnymi šablónami

ETS pracuje s dvoma elementmi: strom dát a šablóny. Strom dát obsahuje všetky dáta, ktoré sú dostupné. Šablóny definujú pôsob, akým budú dáta zobrazené. Tento systém práce ide prirovnať k XML dokumentu transformovanému XSLT šablónami.

Týmto spôsobom je celá prezenčná vrstva uložená v TPL súboroch. Riadiacu časť obsahujú skripty PHP.

### **3.3 OCPS Manager**

OCPS Manager je samostatná časť napísaná v jazyku PHP. Je to z dôvodu používania spoločného kódu s OCPS Webovým serverom. Spúšťa sa ako PHP CLI skript a k behu potrebuje operačný systém Linux.

Tak ako je vidieť na diagrame 3-1, tento komponent zabezpečuje prepojenie vyhodnocovacích serverov s databázou, kde sú uložené požiadavky na vyhodnocovanie. Slúži teda ako organizátor práce. Jeho úlohou je vyberať požiadavky, nájsť voľné prostriedky k vyhodnoteniu požiadavku, teda voľný evaluátor. Manager musí byť schopný zabezpečiť dodanie potrebných dát k evaluátoru, jeho spustenie, následný zber výsledkov a aktualizovanie údajov v databáze na základe získaných dát.

Z podstaty jeho činnosti je zrejmé že bude bežať v nekonečnej slučke. Počas jedného behu by mal vykonať sériu úkonov popisovanú vyššie.

### 3.3.1 Cyklus OCPS Managera

Na začiatku cyklu Manager skontroluje v tabuľke požiadaviek, či obsahuje nejaké záznamy. Pokiaľ je tomu tak skontroluje dostupnosť evaluátorov. Každý evaluátor má v databáze položku, časové razítko, ktorá hovorí o tom, či je voľný. Pokiaľ je časové razítko nulové evaluátor sa môže použiť. Nasleduje skontrolovať dostupnosť počítača na ktorom evaluátor sídli.

V tomto momente Manager vytvorí synovský proces, ktorý ďalej preberie riadenie procesu vyhodnocovania. Sám sa uspí na určitú časovú dobu, aby nezaťažoval operačný systém. Táto doba sa dá nastaviť v konfiguračnom súbore a je dobré nastaviť ju podľa frekvencie zasielania požiadaviek.

Pokiaľ Manager nenájde žiadnu požiadavku k vyhodnocovaniu alebo nie je k dispozícii žiaden voľný evaluátor, dobehne nakoniec, na určitú dobu sa uspí a znova pokračuje v ďalšom cykle. Na konci cyklu ešte prejde všetky svoje synovské procesy, ktoré boli vytvorené v predchádzajúcich behoch a pokiaľ už dobehli sú v systéme ako takzvané zombie procesy, preto ich ukončí.

Vráťme sa teraz k vytvorenému synovskému procesu, ktorý vznikol po splnení podmienky voľného evaluátora a získaného požiadavku k vyhodnoteniu.

Pri preberaní úlohy z tabuľky požiadaviek je potrebné označiť, že úloha sa práve vyhodnocuje. Nastaví sa teda časové razítko začiatku vyhodnocovania na aktuálny čas. Tento údaj je potrebný pre OCPS Webový server, keď zisťuje stav požiadavku.

Pokiaľ chceme vyhodnotiť zaslanú úlohu musíme sa dostať ku konkrétnemu súboru so zdrojovým kódom. Aby bolo jednoznačné určiť daný súbor OCPS server vygeneruje pri získaní súboru od užívateľa unikátny názov súboru. Ten je tvorený podľa schémy:

```
{login}_w{taskId}_{submitTimestamp}.{taskExtension}
```

V konfiguračnom súbore je uložená cesta, kam sa ukladajú riešenia. Dané riešenie sa potom uloží na určené miesto. Vytvorí sa však ešte podadresárová štruktúra, ktorá slúži už len pre lepšiu orientáciu človeka v repozitári. Pre každého užívateľa je tu zvlášť vytvorený adresár a v ňom podadresáre

obsahujúce riešenia konkrétneho problému. Súbory majú natoľko unikátne meno a dostatočne čitateľný názov na to, aby mohli byť všetky v jednom adresári, avšak takto je prípadná manuálna manipulácia pohodlnejšia.

Rovnako dôležité sú i konfiguračné súbory evaluátoru pre danú úlohu. Tie musia byť k dispozícii pre každú aktívnu úlohu na mieste určenom v konfiguračnom súbore OCPS.

Takto získané súbory sa pomocou *scp*, kopírovania súborov na vzdialený počítač, premiestnia do požadovaných adresárov pre evaluátor.

Aby bolo možné komunikovať so vzdialeným počítačom cez ssh a pracovať na hostiteľskom počítači ako iný užívateľ, a to bez nutnosti zadávať heslo, je potreba vygenerovať kľúče, pomocou ktorých sa manager bude identifikovať u evaluátoru.

Všetky dáta sú k dispozícii a pripravené v hodnoteníu. Cez ssh sa spustí evaluátor. Vyprodukované výsledky sa musia znova preniesť na stroj kde sídli Manager. Stačí log súbor z vyhodnocovacieho procesu a súbor obsahujúci verdikt vyhodnocovania. Tie sa pod rovnakými názvami (jednoznačne vygenerovanými) uložia k zaslanému riešeniu.

Zo súborov sa získajú potrebné informácie vytvorí sa nový záznam vo výsledkovej tabuľke, ktorá slúži ako zdroj informácií pre riešiteľov.

Nakoniec je potreba odstrániť záznam z tabuľky požiadaviek, označiť v tabuľke evaluátor ako voľný a tým je proces vyhodnocovania ukončený.

Ako bolo spomínané, manager beží v nekonečnej slučke. Ukončuje sa klasicky zaslaním kill signálu. Vtom prípade ale musí byť zabezpečené aby sa evaluátor neukončil uprostred cyklu. Preto je tu implementovaný zachytávač systémových signálov, ktorý kontroluje vyslané signály. Pokiaľ je vyslaný signál na ukončenie, nastaví sa do premennej, že je potreba skončiť. Aplikácia si v inej premennej uchováva, kedy môže byť ukončená a pokiaľ je táto hodnota nastavená kladne a bola zaznamenaná požiadavka na ukončenie, aplikácia sa ukončí. Funkcia, ktorá zabezpečuje kontrolu oboch premenných sa spúšťa každú sekundu.

### 3.4 Evaluátor

Evaluátor, inak nazývaný tiež Judge, je výkonnou časťou aplikácie. Jeho úloha je otestovať riešenie problému. Ako taký evaluátor funguje je popísané úvodnej časti práce.

Ja som vlastný evaluátor nevytváral. Nakoniec to ani nebolo cieľom tejto práce. Použil som MO-Eval [10], ktorý napísal Matin Mareš. Ide o systém, ktorý sa používa ako offline validátor na olympiádach z informatiky.

Tento systém bol bez zmeny vhodný k zabudovaniu do OCPS. Ako offline validátor môže byť inštalovaný na hocikáký počítač. Pokiaľ dodržíme jeho konvencie, uložíme mu vyhodnocovanú úlohu tak ako to má nastavené, nahráme patričné konfiguračné súbory a spustíme pomocou OCPS Managera vyhodnotenie úlohy na diaľku, potom zozbierame výsledky, nič nebráni tomu aby bol systém efektívny.

Pokiaľ sú konfiguračné súbory správne zostavené a uložené na presnom mieste tak by ich OCPS Manager mohol distribuovať, môžeme zároveň použiť niekoľko evaluátorov zároveň. Je však podmienkou, aby každý Judge bol nainštalovaný na inom počítači. Aby bolo viac týchto evaluátorov na jednom stroji nedáva žiaden význam, pretože dva systémy sa i tak musia o procesor deliť. Druhým argumentom je čas vyhodnotenia jednej úlohy. Je tak malý, že používaním viac evaluátorov na jednom počítači by neprinieslo žiadne zrýchlenie a nemá teda ani význam o tom uvažovať. Naopak rozdistribution evaluátorov na rôzne počítače zabezpečí zvýšenie výkonu pri vyhodnocovaní a zároveň poistí chod aplikácie v prípade technického výpadku stroja, kde jeden z evaluátorov beží.

## **4 UŽÍVATEĽSKÁ DOKUMENTÁCIA**

### **4.1 Systémové požiadavky**

#### **4.1.1 Server**

- Operačný systém Linux s jadrom 2.4
  - správna funkčnosť vyhodnocovacieho servera závisí na jadre 2.4 a i386
- Databázový server MySQL verzie 4 (alebo novšej)
- Webový server
  - Apache 2.2 (staršie verzie Apache neboli testované)
- PHP verzie 5.2.1 (alebo novšie)
- Rozšírenia PHP
  - mysql
  - sockets
  - pnctI

#### **4.1.2 Klient**

Pre spustenie aplikácie stačí iba webový prehliadač. Doporučené prehliadače:

- Mozilla Firefox 1.5 (a novší)
- Internet Explorer 6, nebo 7

## **4.2 Inštalácia**

### **4.2.1 Webový server**

Aplikácia OCPS vyžaduje pre svoj beh webový server Apache 2.2.

Webový server Apache verzie 2.2.4 bude kompilovaný s podporou mod\_access, mod\_auth, mod\_auth\_digest, mod\_deflate, mod\_env, mod\_headers, mod\_setenvif, mod\_ssl, mod\_mime, mod\_imap, mod\_alias a mod\_rewrite. Pokiaľ nie je uvedené inak, predpokladá sa, že všetky zdrojové súbory sú uložené v adresári /usr/local/src.

## Kompilácia (open) SSL

Ešte než sa začne s kompiláciou samotného Apache, je nutné skompilovať SSL. K tomu je potrebná knižnica OpenSSL (*openssl-0.9.8e.tar.gz*). Nachádza sa na inštalačnom CD aplikácie OCPS alebo na adrese:

<http://www.openssl.org/source/>.

1. Kompilácia OpenSSL vypadá nasledovne:

```
# cd /usr/local/src
# tar -zxvf openssl-0.9.7d.tar.gz
# cd openssl-0.9.7d
# ./config --prefix=/usr/local
# make
# make install
```

2. Vygenerovanie privátneho kľúča do adresára /home/ssl:

```
# mkdir /home/ssl
# cd /home/ssl
/usr/local/bin/openssl genrsa -des3 -rand \
/usr/local/bin/openssl genrsa -des3 -rand \
/usr/local/bin/openssl genrsa -des3 -rand \
/usr/local/bin/openssl genrsa -des3 -rand \
-out localhost.key 1024
```

3. Následne sa vygeneruje privátny kľúč bez pass-phrase. Je to síce menej bezpečné, ale nie je potom potreba zadávať pass-phrase pri každom bootupu webového serveru:

```
# /usr/local/bin/openssl rsa \
-in localhost.key \
-out localhost.key.unsecure
```

4. Ďalej sa vytvorí požiadavka na certifikačnú autoritu (localhost.key.csr), ktorá kľúč autorizuje:

```
# /usr/local/bin/openssl req -new \  
    -key localhost.key \  
    -out localhost.key.csr
```

## Kompilácia Apache 2.2.4 webového serveru

Ku kompilácii sú potrebné zdrojové kódy Apache. Je to súbor *httpd-2.2.4.tar.bz2*. Nachádza sa na inštalačnom CD aplikácie OCPS alebo je možné ho stiahnuť z adresy:

<http://httpd.apache.org/download.cgi>

1. Nasledujúce príkazy spustia kompiláciu Apache (spúšťajte ako užívateľ *root*):

```
# cd /usr/local/src  
# tar -zxvf httpd-2.0.51.tar.gz  
# cd httpd-2.0.51  
# ./configure \  
    --prefix=/usr/local/apache2 \  
    --enable-so \  
    --enable-auth-digest \  
    --enable-rewrite \  
    --enable-setenvif \  
    --enable-mime \  
    --enable-deflate \  
    --enable-ssl \  
    --with-ssl=/usr/local \  
    --enable-headers  
# make  
# make install
```

2. Rovnako je nutné modifikovať (doplniť) konfiguračný súbor Apache */usr/local/apache2/conf/httpd.conf*. (tento príklad predpokladá, že *dokument root* je v */home/web*):

```
DocumentRoot "/home/www"
```



Doplní sa PHP 5, ktoré bude inštalované ako modul Apache 2 serveru:

```
LoadModule php5_module modules/libphp5.so
DirectoryIndex index.html index.htm index.php
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

Doplnia sa základné pravidlá pre mod\_rewrite:

```
<Directory "/home/www">
    Options Indexes Includes FollowSymLinks MultiViews
    AllowOverride All
    Order allow,deny
    Allow from all
</Directory>
<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
</Files>
```

3. Pre použitie SSL (na štandardnom porte 443) sa zmení obsah súboru  
/usr/local/apache2/conf/ssl.conf na:

```
Listen 443
<VirtualHost _default_:443>
    ServerName localhost
    SSLEngine on
    SSLCertificateFile /home/ssl/localhost.cert
    SSLCertificateKeyFile /home/ssl/localhost.key.unsecure
</VirtualHost>
```

## Inštalácia PHP 5.2 zo zdroja

Archív zo zdrojovými kódmi PHP (*php-5.2.2.tar.bz2*) je umiestnený na inštalačnom CD aplikácie OCPS alebo k dispozícii k stiahnutiu na adrese <http://www.php.net/downloads.php>.

Pod užívateľom *root* sa vykonajú nasledujúce príkazy:

```
# cd /usr/local/src  
  
# tar -jxvf php-5.2.2.tar.bz2  
  
# cd php-5.2.2.tar.bz2
```

V závislosti na požiadavkách je možné v nasledujúcom príkaze zapnúť alebo vypnúť rôzne funkcie PHP doplnením či odobraním prepínačov. Viac informácií a presné znenie prepínačov je možné získať po zadaní `./configure --help`.

Ďalej sa pokračuje:

```
# ./configure \  
--with-apxs2=/usr/local/apache2/bin/apxs \  
--with-mysql=/usr/local/mysql \  
--with-mysql-sock=/tmp/mysql.sock \  
--with-gettext \  
--with-config-file-path=/usr/local/apache2/conf \  
--with-openssl \  
--enable-debug=yes \  
--enable-sockets \  
--enable-pcntl
```

Ďalej sa pokračuje:

```
# make  
  
# make install  
  
# cp php.ini-dist /usr/local/apache2/conf/php.ini
```

Po nainštalovaní php je nutné doplniť premenné prostredia a reštartovať server Apache. Poslúži k tomu nasledujúci skript:

```
#!/bin/sh  
  
APACHEHOME=/usr/local/apache2  
  
echo Starting Apache
```

```
$APACHEHOME/apachectl start
```

## Ištalácia databázového servera MySQL

MySQL databáza je používaná vo verzii 4.1 postup pre vyššie verzie by bol totožný. Je možné si nainštalovať i binárnu verziu z balíčku príslušnému k vašej distribúcii Linuxu.

Príslušné inštalačné či zdrojové súbory sú na inštalačnom CD alebo si ich je možné stiahnuť na adrese: <http://dev.mysql.com/downloads/mysql/4.1.html>.

- Vlastná Mysql 4.1.22 databáza - [mysql-4.1.22.tar.gz](#)

Kompilácia MySQL včetně vytvorenia skupiny a užívateľa mysql:

```
# cd /usr/local/src
# tar -zxvf mysql-4.0.21.tar.gz
# cd mysql-4.0.21
# ./configure --prefix=/usr/local/mysql
# make
# make instar
# groupadd mysql
# useradd -g mysql mysql
# cp support-files/my-medium.cnf /etc/my.cnf
# cd /usr/local/mysq
# bin/mysql_install_db --user=mysql
# chown -R root .
# chown -R mysql var
# chgrp -R mysql .
# bin/mysqld_safe --user=mysql
```

Soubor /etc/my.cnf môže byť pre malú záťaž MySQL databáze konfigurovaný napríklad takto:

```
[client]
port = 3306
```

```
socket = /tmp/mysql.sock

[mysqld]
port = 3306
socket = /tmp/mysql.sock
skip-locking
key_buffer = 16K
max_allowed_packet = 1M
table_cache = 4
sort_buffer_size = 64K
net_buffer_length = 2K
thread_stack = 64K
skip-networking
server-id = 1

[mysqldump]
quick
max_allowed_packet = 16M

[mysql]
no-auto-rehash

[isamchk]
key_buffer = 8M
sort_buffer_size = 8M

[myisamchk]
key_buffer = 8M
sort_buffer_size = 8M

[mysqlhotcopy]
interactive-timeout
```

## phpMyAdmin

- Inštalačný balíček je priložený na inštalačné CD aplikácie OCPS alebo si ho stiahneme: [phpMyAdmin-2.11.1-all-languages.tar.bz2](http://phpmyadmin.sourceforge.net/phpMyAdmin-2.11.1-all-languages.tar.bz2)

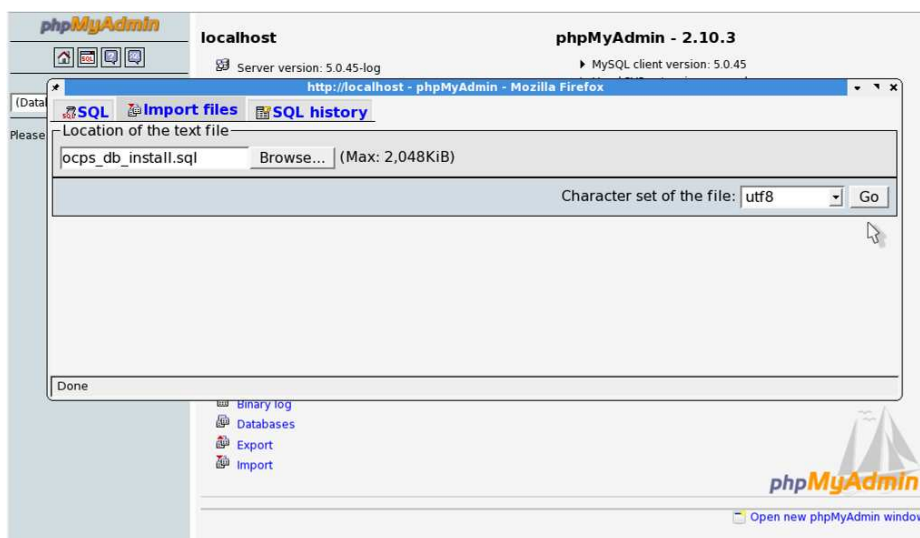
phpMyAdmin sa rozbalí pomocou príkazu `tar -jxvf do /home/www`. Za pomoci phpMyAdminu je možné spravovať MySQL databázu pomocou webového rozhrania. Tento nástroj bude slúžiť na prípadnú konfiguráciu aplikácie OCPS.

## Inštalácia OCPS

### **OCPS server**

1. Obsah súboru *OCPS\_web.tar.bz2* sa rozbalí do adresára, kde má webový server uložené aplikácie. Pokiaľ je nainštalovaný webový server Apache, tak je to štandardne `/home/www`
2. Je potreba nastaviť práva pre adresár `/upload` tak, aby do neho mal webový server prístup i pre zápis. Všetky ostatné adresáre môžu mať prístup len pre čítanie.
3. Upraví sa nastavenie v súbore */\_settings.php* (podrobnejšie informácie sú v sekcii Konfigurácia)

4. V adresári *install* sa nachádza súbor *ocps\_db\_install.sql*. Je potrebné



zmeniť obsah súboru. A to konkrétne na riadku 9 sa upraví heslo pre prístup užívateľa *ocps*.

V aplikácii phpMyAdmin po kliknutí na ikonku *SQL*, zvolíme *Import files*, vyššie spomínaný súbor vyberieme a aplikujeme.

#### 4-1: inštalácia databázy pre OCPS

5. Pre potreby *OCPS manager* sa vytvorí užívateľ *ocps*:
- ```
# useradd -d /aux/ocps \  
-g 35000 -u 35000 -s /bin/bash -c "OCPS Manager" ocps
```
6. Pre vytvorenie adresárovej štruktúry skopírovanie súborov spustíme ako užívateľ *root*:
- ```
# install/manager-isntall
```

#### **Inštalácia vyhodnocovacieho serveru JUDGE**

*JUDGE* môže byť nainštalovaný na fyzicky inom počítači ako *webový server OCPS* a *databázový server MySQL*. Môže byť dokonca súčasne viac severov *JUDGE*, ale každý na inom stroji.

1. Obsah súboru *OCPS\_judge.tar.bz2* sa rozbalí na ľubovoľné miesto.
2. Spustí sa príkaz pre skompilovanie:  
  

```
#make
```
3. Upraví sa súbor *config* podľa našich požiadaviek, štandardne sa nastaví *MO\_ROOT*, *EVAL\_USER*, *EVAL\_GROUP*

odporúča sa nechať nastavenia v pôvodnom stave, inak je potreba v ďalších krokoch upraviť postup podľa vykonaných zmien v súbore *config*.

4. Vytvorí sa užívateľ:

```
mo-eval:x:65000:65000:MO                               Evaluator:/aux/mo/eval/mo-  
eval:/bin/bash
```

Toto je potrebné vykonať ako užívateľ *root* príkazom:

```
# useradd -d /aux/mo/eval/mo-eval \  
-g 65000 -u 65000 -s /bin/bash -c "MO Evaluator" mo-eval
```

Vytvoríme heslo pre užívateľa *mo-eval*:

```
#passwd mo-eval
```

5. Vytvorí sa skupina *mo-eval* (Toto je potrebné vykonať ako užívateľ *root*.):

```
# groupadd -g 65000 mo-eval .
```

6. Vytvorí sa súborová štruktúra pod *\$MO\_ROOT* a nainštalujú sa všetky časti vyhodnocovacieho servera a nastaví sa potrebné práva (Ako užívateľ *root*.):

```
# bin/mo-install
```

## 4.3 Konfigurácia

### 4.3.1 Konfigurácia OCPS webového serveru

Subor *\_settings.php* obsahuje konfiguračné položky, ktoré je možné nastaviť.

Nastavenia sa odporúčajú ponechať v pôvodnom nastavení. Pokiaľ sa nejaké nastavenia zmenia môže to znefunkčniť aplikáciu, pretože nastavenia sú zviazané s inými komponentmi aplikácie

Premenná	Význam
<code>\$cfg['db']['host']</code>	Server, kde je umiestnená databáza MySQL
<code>\$cfg['db']['user']</code>	Užívateľ v databázy MySQL, musí byť rovnaký ako v inštalačnom skripte (štandardne <i>ocps</i> )

<b>\$cfg['db']['passwd']</b>	Heslo užívateľa <i>ocps</i> aké sa nastavovalo v inštalačnom skripte v sekcii 3.1.6 v kroku 4.
<b>\$cfg['db']['db_name']</b>	Názov databázy. Štandardne je to <i>ocps</i>
<b>\$cfg['blowfish_key']</b>	<p>Tajný kľúč, ktorý sa používa na šifrovanie. Kľúč môže obsahovať ľubovoľnú postupnosť znakov a môže byť dlhý najviac 56 znakov.</p> <p><b>Hodnota kľúče by mala byť zmenená ihneď po inštalácii aplikácie</b></p>
<b>\$cfg['session_exp_in_minutes']</b>	<p>Určuje životnosť session v minútach. Pokiaľ užívateľ neprevedie žiadnu akciu behom zadanej doby, bude automaticky odhlásený.</p> <p>Pokiaľ bude hodnota nastavená na 0 (nulu), potom nedôjde k automatickému odhláseniu.</p>
<b>\$cfg['stylesheet']</b>	Súbor štýlov vzhľadu pre webové rozhranie aplikácie.
<b>\$cfg['ocps']['upload_dir']</b>	Adresár pre dočasné uloženie nahraných súborov na server
<b>\$cfg['ocps']['problems']</b>	<p>Adresár obsahujúci konfiguračné súbory pre <i>vyhodnocovací server ocps</i> pre jednotlivé úlohy.</p> <p>Musí byť umiestnený na serveri, kde <i>webový server ocps</i>.</p>
<b>\$cfg['ocps']['solutions']</b>	<p>Adresár obsahujúci zaslané riešenia a výstupné súbory evaluátora.</p> <p>Musí byť umiestnený na serveri, kde je <i>webový server ocps</i>.</p>
<b>\$cfg['evaluator']['problems']</b>	<p>Adresár <i>vyhodnocovacieho serveru ocps</i> obsahujúci konfiguračné súbory jednotlivých úloh.</p> <p>Musí byť umiestnený na serveri, kde <i>vyhodnocovací server ocps (JUDGE)</i>. Je jeho súčasťou.</p>



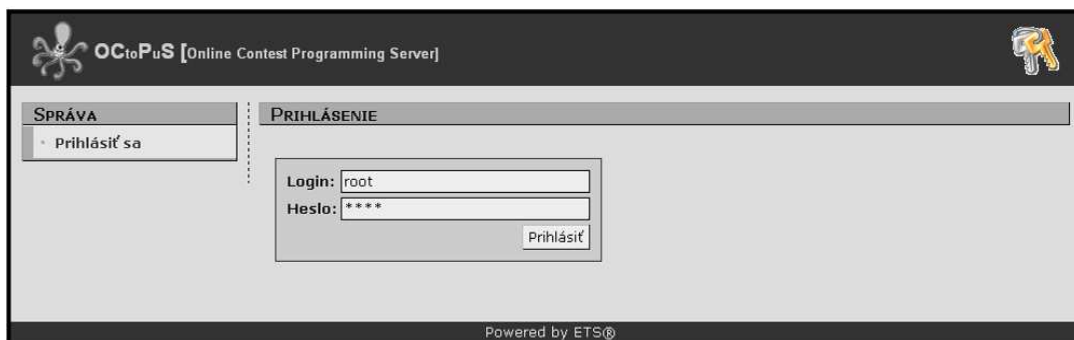
<b>\$cfg['evaluator']['solutions']</b>	<p>Adresár <i>vyhodnocovacieho serveru ocps</i> obsahujúci zaslané riešenia vyhodnoteniu <i>vyhodnocovaciemu serveru ocps</i> a je ho výstupné súbory.</p> <p>Musí byť umiestnený na serveri, kde <i>vyhodnocovací server ocps (JUDGE)</i>. Je jeho súčasťou.</p>
<b>\$cfg['evaluator']['testing']</b>	<p>Adresár <i>vyhodnocovacieho serveru ocps</i> obsahujúci vyhodnocovanú úlohu a výsledky z tohto procesu.</p> <p>Musí byť umiestnený na serveri, kde <i>vyhodnocovací server ocps (JUDGE)</i>. Je jeho súčasťou.</p>
<b>\$cfg['manager_round_interval']</b>	<p>Časová perióda v sekundách, ako často OCPS Manager kontroluje tabuľku požiadaviek na vyhodnocovanie.</p>
<b>\$cfg['summary_groups_row_per_table']</b>	<p>Označuje počet riadkov tabuľky pre zoznam výsledkov pre jednotlivých užívateľov v skupine. Táto tabuľka je k dispozícii a administratívnej sekcii v kategórii sumarizácií.</p> <p>Pre lepšiu prehľadnosť bude zoznam riešiteľov s skupine rozdelený do tabuliek s počtom riadkov odpovedajúcim hodnote položky.</p>

Nastavenia jazykových variant musia ostať ponechané bezo zmeny.

### 4.3.2 Administrácia obsahu




Administrácia obsahu je vykonávaná cez webové rozhranie aplikácie OCPS. Do administratívneho rozhrania sa dostaneme z prihlasovacej stránky k aplikácii. Pokiaľ sa prihlásený užívateľ disponuje právami administrátora je automaticky resmerovaný do administratívnej časti aplikácie.

Automaticky po nainštalovaní aplikácie existuje absolútny administrátor *root*. Jeho heslo je nastavené na hodnotu „*ocpsroot*“. Po prvom prihlásení sa do aplikácie pod týmto užívateľom sa odporúča toto heslo zmeniť.



#### 4-1: OCPS - prihlasovacia stránka

Každá entita (užívateľ, skupina, problém, ...) má tri ovládacie prvky. Pomocou nich je možné vykonávať operácie s entitou.

Symbol	Názov	Význam
	View	Zobrazí informácie o entite
	Edit	Umožní upravovať parametre entity
	Delete	Zmaže entitu

Na vrchu každej stránky sa nachádzajú ovládacie prvky, ktoré nám umožňujú manipuláciu s entitami. Na súhrnnej stránke je to odkaz na vytvorenie novej entity. Na stránke s informáciami o entite je to odkaz na editáciu a zmazanie entity, u entity *užívateľ* je to navyše zmena hesla. Na stránke pre editáciu entity je to odkaz na zmazanie. Stránka pre zmazanie už neobsahuje žiadne ovládacie prvky.

Niektoré ovládacie prvky môžu v istých situáciách chýbať. Je to zapríčinené tým, že administrátor nemá k danej akcii oprávnenie. Typickým príkladom je zmazanie vlastného účtu.

### Skupiny užívateľov

Administráciu skupín užívateľov vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam skupín.

**OCPS [Online Contest Programming Server]**

**PRIHLÁSENÝ ...**  
Užívateľ: root

**ADMINISTRÁCIA - SKUPINY**  
[Pridať skupinu](#)

#	Názov	Poznámka	Dátum vytvorenia	+	-	?	!
1	Alfa Group	Pre najnáročnejšie úlohy	2007-02-28 17:11:29	I	Q	!	X
2	Beta Group	Skupina strednej náročnosti	2007-02-28 17:47:51	I	Q	!	X
3	Gamma Group	Skupina ľahkej náročnosti	2007-11-20 00:45:26	I	Q	!	X

Powered by ETS@

## 4-2: OCPS – skupiny užívateľov

### Informácie o skupine užívateľov

Po vstupe na stránku sa zobrazia informácie o skupine. Okrem týchto informácií je tu zoznam užívateľov, ktorí patria do skupiny a tiež zoznam úloh prislúchajúcich ku skupine. Pri úlohách je možné priamo odoberať úlohu, alebo pridať novú.

### Pridanie skupiny užívateľov

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je názov skupiny.

Ďalej je možné rovno zvoliť užívateľov, ktorí budú do skupiny patriť.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nová skupina.

### Editovanie skupiny užívateľov

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie, informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

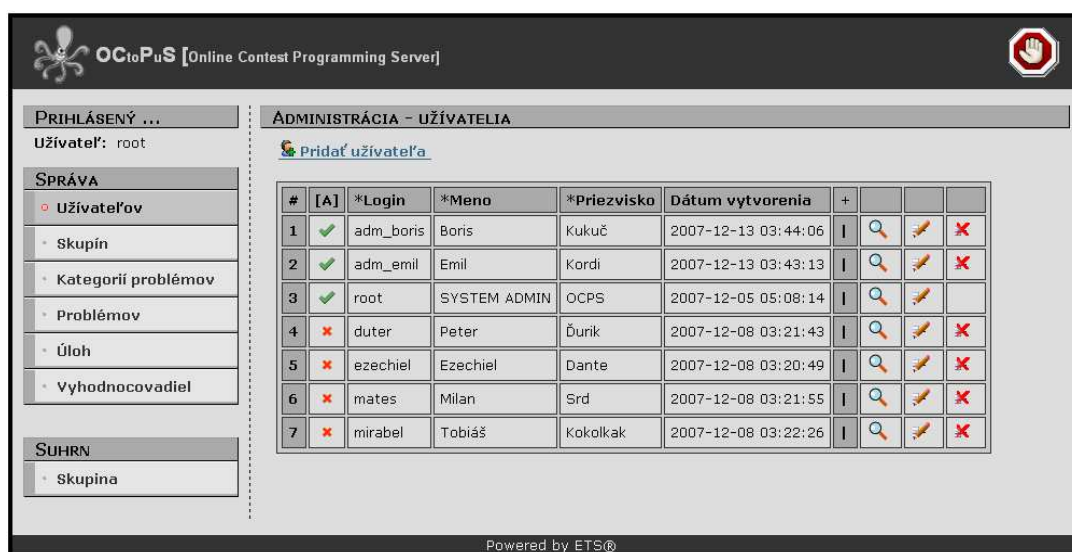
### Zmazanie skupiny užívateľov

Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity s informáciou, o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je skupina zmazaná a rovnako sú zmazané všetky vzťahy príslušnosti užívateľov k tejto skupine.

## Užívatelia

Administráciu užívateľov vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam užívateľov. Užívatelia s administrátorskými oprávneniami sú označený zeleným symbolom.



#	[A]	*Login	*Meno	*Priezvisko	Dátum vytvorenia	+			
1	✓	adm_boris	Boris	Kukuč	2007-12-13 03:44:06		🔍	✏️	✖️
2	✓	adm_emil	Emil	Kordi	2007-12-13 03:43:13		🔍	✏️	✖️
3	✓	root	SYSTEM ADMIN	OCPS	2007-12-05 05:08:14		🔍	✏️	
4	✗	duter	Peter	Đurik	2007-12-08 03:21:43		🔍	✏️	✖️
5	✗	ezechiel	Ezechiel	Dante	2007-12-08 03:20:49		🔍	✏️	✖️
6	✗	mates	Milan	Srd	2007-12-08 03:21:55		🔍	✏️	✖️
7	✗	mirabel	Tobiáš	Kokolkak	2007-12-08 03:22:26		🔍	✏️	✖️

### 4-3: OCPS – užívatelia

## Informácie o užívateľovi

Po vstupe na stránku sa zobrazia informácie o užívateľovi. Okrem týchto informácií je tu zoznam skupín, kam užívateľ patrí. Toto sa netýka užívateľov s administrátorskými právami. Oni nemajú definovanú príslušnosť ku skupine.

### Pridanie užívateľa

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je *login*. Nastavovať administrátorské práva novému užívateľovi má iba *root*.

Ďalej je možné rovno zvoliť skupiny, kam bude užívateľ patriť. Ak je nový užívateľ označený ako administrátor, toto nastavenie sa neberie v úvahu.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nový užívateľ.

### ***Editovanie užívateľa***

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Zmeny osobných údajov a príslušnosť ku skupinám sa nastavujú zvlášť. Pokiaľ je aktuálne editovaný užívateľ administrátor, možnosť editovať príslušnosť ku skupinám nie je k dispozícii.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

### ***Zmazanie užívateľa***



Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity s informáciou, o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je užívateľ zmazaný a rovnako sú zmazané všetky vzťahy príslušnosti užívateľa ku skupinám.

## **Kategórie problémov**

Administráciu kategórií problémov vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam kategórií.

Kategórie môžu byť usporiadané do stromovej štruktúry. Tie, ktoré sú úplne na konci vetiev, môžu obsahovať problémy. Sú v zozname označené symbolom listu.


**OCtoPuS** [Online Contest Programming Server]
 

**PRIHLÁSENÝ ...**  
 Užívateľ: root
















**ADMINISTRÁCIA – KATEGORIE**  
[Pridať kategóriu](#)

**SPRÁVA**

- Užívateľov
- Skupín
- Kategórií problémov**
- Problémov
- Úloh
- Vyhodnocovadiel

**SUHRN**

- Skupina

#	[+]	Názov	Poznámka	+			
1.		`- contest	Obsahuje problémy vhodné pre súťaže				
2.		`- testing	kategória obsahujúca cvičné úlohy				
3.		`- algo	skupina s obsahom precvičovania algoritmov				
4.		`- speed	kategória problémov zameraných na precvičovanie optimalizácie rýchlosti programu				

Powered by ETS®

#### 4-4: OCPS – kategórie problémov

##### **Informácie o kategóriách problémov**

Po vstupe na stránku sa zobrazia informácie o kategóriách problémov. Okrem týchto informácií je tu zoznam kategórií, ktoré patria hierarchicky pod danú kategóriu alebo zoznam úloh prislúchajúcich do kategórie pokiaľ sa jedná o kategóriu na konci hierarchie.

##### **Pridanie kategórie problémov**

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je názov kategórie.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nová kategória problémov.

##### **Editovanie kategórie problémov**

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

### Zmazanie kategórie problémov

Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity, s informáciou o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je kategória problémov zmazaná.

Mazať kategóriu je možné len v prípade, že je na konci hierarchie a neobsahuje žiadne problémy.

## Problémy

Administráciu problémov vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam problémov.

OCPS [Online Contest Programming Server]

PRIHLÁSENÝ ...  
Užívateľ: root

ADMINISTRÁCIA - PROBLÉMY

[Pridať problém](#)

#	Názov	Poznámka	Dátum vytvorenia				
1	Problem A	Firetruck	2007-12-13 05:06:27	I	🔍	✏️	✖️
2	Problem B	SameGame Simulation	2007-12-13 05:06:47	I	🔍	✏️	✖️
3	Problem C	Concurrency Simulator	2007-12-13 05:07:06	I	🔍	✏️	✖️
4	Problem D	The Domino Effect	2007-12-13 05:32:45	I	🔍	✏️	✖️
5	Problem E	Processing MX Records	2007-12-13 05:32:56	I	🔍	✏️	✖️
6	Problem F	Pseudo-Random Numbers	2007-12-13 05:33:07	I	🔍	✏️	✖️

Powered by ETS®

### 4-5: OCPS – problémy

#### Informácie o probléme

Po vstupe na stránku sa zobrazia informácie o probléme. Je tu taktiež možnosť vytvoriť priamo z daného problému úlohy ku konkrétnej skupine.

#### Pridanie problému

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je názov problému.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nový problém.

### Editovanie problému

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

### Zmazanie problému

Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity, s informáciou o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je problém zmazaný.

## úlohy

Administráciu úloh vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam úloh.

**ADMINISTRÁCIA - ÚLOHY**

[Pridať úlohu](#)

#	Skupina	Problém	Začiatok	Koniec				
1	Alfa Group	Problem A	2007-10-12 13:00:40	2008-12-04 13:00:00		🔍	✏️	✖️
2	Alfa Group	Problem B	2007-10-12 13:00:00	2008-12-04 13:00:00		🔍	✏️	✖️
3	Alfa Group	Problem D	2007-03-12 13:00:00	2008-12-04 13:00:00		🔍	✏️	✖️
4	Alfa Group	Problem E	2007-03-12 13:00:00	2008-12-04 13:00:00		🔍	✏️	✖️
5	Alfa Group	Problem F	2007-03-12 13:00:00	2008-08-22 13:00:00		🔍	✏️	✖️
6	Beta Group	Problem C	2007-03-12 13:00:00	2007-12-12 18:45:59		🔍	✏️	✖️
7	Beta Group	Problem D	2007-10-12 18:45:00	2007-12-12 12:00:00		🔍	✏️	✖️
8	Beta Group	Problem E	2007-11-01 15:00:00	2007-12-05 16:00:00		🔍	✏️	✖️
9	Gama Group	Problem A	2007-03-21 13:00:40	2008-12-04 13:00:00		🔍	✏️	✖️
10	Gama Group	Problem D	2007-11-12 18:45:46	2008-01-05 13:00:00		🔍	✏️	✖️
11	Gama Group	Problem E	2007-10-12 18:45:00	2007-11-30 12:00:00		🔍	✏️	✖️
12	Gama Group	Problem F	2007-10-12 18:45:00	2007-11-04 18:45:59		🔍	✏️	✖️

Powered by ETS®

### 4-6: OCPS – úlohy

### Informácie o úlohe

Po vstupe na stránku sa zobrazia informácie o úlohe.



### ***Pridanie úlohy***

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je v tomto prípade existencia úlohy vytvoreného z rovnakého problému v skupine.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nová úloha.

### ***Editovanie úlohy***

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

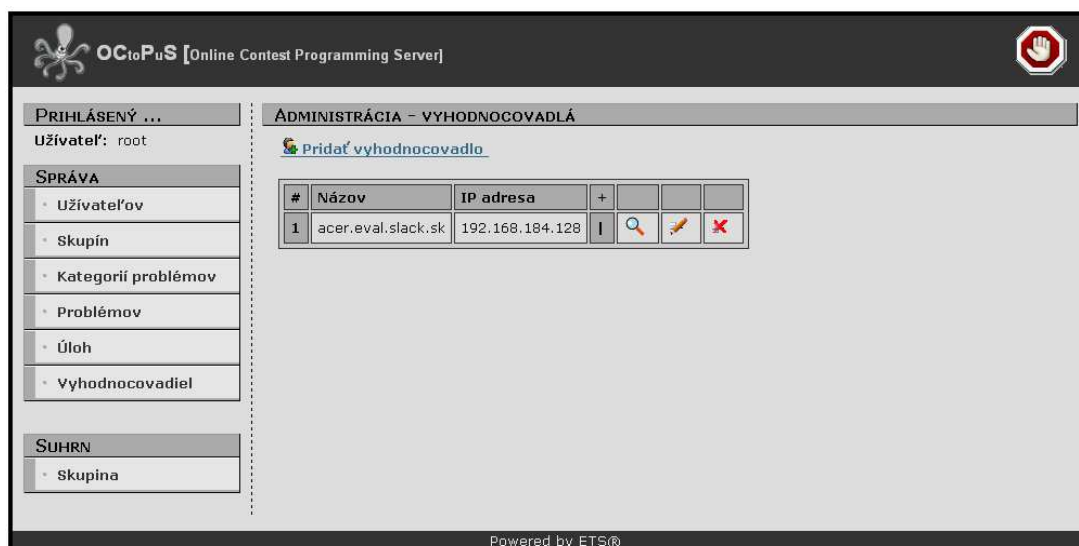
### ***Zmazanie úlohy***

Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity, s informáciou o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je úloha zmazaná.

## **Evaluátory (vyhodnocovadlá)**

Administráciu evaluátorov vyvoláme cez príslušnú položku v menu.

Po vstupe do sekcie sa zobrazí aktuálny zoznam evaluátorov.



#### 4-7: OCPS – evaluátory

##### **Informácie o evaluátore**

Po vstupe na stránku sa zobrazia informácie o evaluátore.

##### **Pridanie evaluátoru**

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Po ich vyplnení sa skontroluje, či boli zadané povinné údaje a či niektoré údaje nie sú v rozpore s pravidlami. Typicky sa jedná o unikátne parametre ako je v tomto prípade IP adresa evaluátoru.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa vytvorí nový evaluátor.

##### **Editovanie evaluátoru**

Po vstupe na stránku sa zobrazí formulár, kde sú polia pre jednotlivé atribúty entity. Obsah formulárových polí je predvyplnený aktuálnymi hodnotami.

Ak sú všetky údaje korektné, po kliknutí na tlačidlo *Aplikovať* sa aplikujú nové nastavenia. V opačnom prípade je zobrazené chybové hlásenie informujúce o konkrétnom probléme. Pri aplikácii sa použije rovnaká kontrola údajov, ako je to v prípade vytvorenia entity.

### **Zmazanie evaluátoru**

Po vstupe na túto stránku sa zobrazí dialóg o potvrdení zmazania entity, s informáciou o akú konkrétnu entitu sa jedná. Pokiaľ potvrdíme zmazanie, je evaluátor zmazaný.

### **4.3.3 OCPS manager**

Konfiguračný súbor *\_settings.php* nachádzajúci sa podľa pôvodnej inštalácie v */aux/ocps* by mal byť totožný s rovnomeným súborom *webového serveru OCPS*.

Vysvetlenie jednotlivých položiek je k dispozícii v kapitole 4.3.1.

### **Prístup k vyhodnocovaciemu serveru JUDGE**

Aby sme zaistili prístup *OCPS managera* k vyhodnocovacím serverom *JUDGE* bez použitia hesla cez *ssh* (čo je bezpochyby nevyhnutné) musíme vygenerovať autentifikačnú dvojicu kľúčov.

Postup rovnaký i v prípade, že *JUDGE* je umiestnený na rovnakom stroji ako *Webový server OCPS*.

To sa vykoná nasledovne:

1. Ako užívateľ *ocps* spustíme príkaz:  

```
# ssh-keygen -t dsa
```
2. Všetky dotazy odklepeme klávesom *ENTER*.
3. Užívateľovi *mo-eval* vytvoríme v domovskom adresári adresár *.ssh*  

```
# ssh mo-eval@{IP-adresa-JUDGE} mkdir -p .ssh
```

Potvrdíme heslom pre užívateľa *mo-eval*.
4. Stále ako užívateľ *ocps* spustíme z domovského adresára príkaz:  

```
# cat .ssh/id_dsa.pub | ssh mo-eval@{IP-adresa-JUDGE} \  
'cat >> .ssh/authorized_keys'
```

Potvrdíme heslom pre užívateľa *mo-eval*.
5. Vyskúšame funkčnosť:  

```
# ssh mo-eval@{IP-adresa-JUDGE}
```

Zalogovať by sa malo podariť bez použitia hesla.

Daný postup sa bude opakovať pre každý *Vyhodnocovací server OCPS* (evaluátor) s tým, že kroky 1. a 2. sa vykonajú len jeden krát a pre ďalšie servery sa pokračuje od kroku 3.

#### 4.3.4 Vyhodnocovací server OCPS (JUDGE)

Vyhodnocovací server *ocps* - *JUDGE* je súčasť aplikácie *OCPS*. Každý vyhodnocovací systém môže mať v danom momente nainštalovaných viac vyhodnocovacích serverov aby sa urýchlila doba vyhodnocovania požiadaviek. Je nutné mať každý vyhodnocovací server nainštalovaný na inom fyzickom počítači.

*JUDGE* podporuje vyhodnocovanie takzvaných *offline* úloh. Ide o úlohy, ktoré načítajú dáta zo štandardného vstupu a vyprodukujú dáta na štandardný výstup. Tento je potom porovnávaný so vzorom a je vyrobený výsledný verdikt.

#### Konfiguračné súbory pre vyhodnocovanie úloh

Pre vyhodnocovací server musia byť k dispozícii konfiguračné súbory pre každú vyhodnocovanú úlohu.

Súbory sú uložené pod adresárovou štruktúrou na rovnakom počítači, ako webový server a MySQL databáza. Je to adresár *problems/{task\_id}*. Mimo iné je tento adresár v konfiguračnej premennej `$cfg['evaluator']['problems']` webového serveru (viď. 4.3.1). Adresár *problems* sa vytvorí inštaláčnou utilitou *manager-install* v kroku 6. postupu inštalácie *OCPS server*. Pre každú úlohu tu musí byť vytvorený podadresár s názvom `{task_id}`, čo je *id* úlohy z tabuľky *tasks*.

Zoznam súborov v danom adresári musí obsahovať:

- `config` – lokálny konfiguračný súbor úlohy (task-u), ktorý prepisuje globálne nastavenia top-level konfiguračného súboru *JUDGE*.
- `{test}.config` – každý testovací prípad má svoj vlastný konfiguračný súbor pre predefinovanie hodnoty atribútu z konfiguračného súboru pre danú úlohu či globálnych nastavení.
- `{test}.in` – testovacie vstupné dáta pre jednotlivé testovacie prípady

- `{test}.out` – očakávané výstupné dáta pre jednotlivé testovacie prípady

POZNÁMKA: `{test}` – bude číslo počnúc číslom 1. Napr. `1.config`, `1.in` a `1.out`.

Podporované sú v tejto chvíli prípony `c`, `C`, `cpp` (pre jazyk C) a `p`, `pas` (pre jazyk Pascal). Na to je potrebné myslieť pri vytváraní úlohy a to konkrétne pri nastavovaní prípony.

## Konfiguračné atribúty súboru *config*

Jednotlivé atribúty sú možné nájsť spolu s popisom v súbore *config* uloženom v adresári vyhodnocovacieho serveru JUDGE (štandardne je to `/aux/mo/eval/mo-eval/config`).

## 4.4 Práca s aplikáciou

### 4.4.1 Užívateľské webové rozhranie.

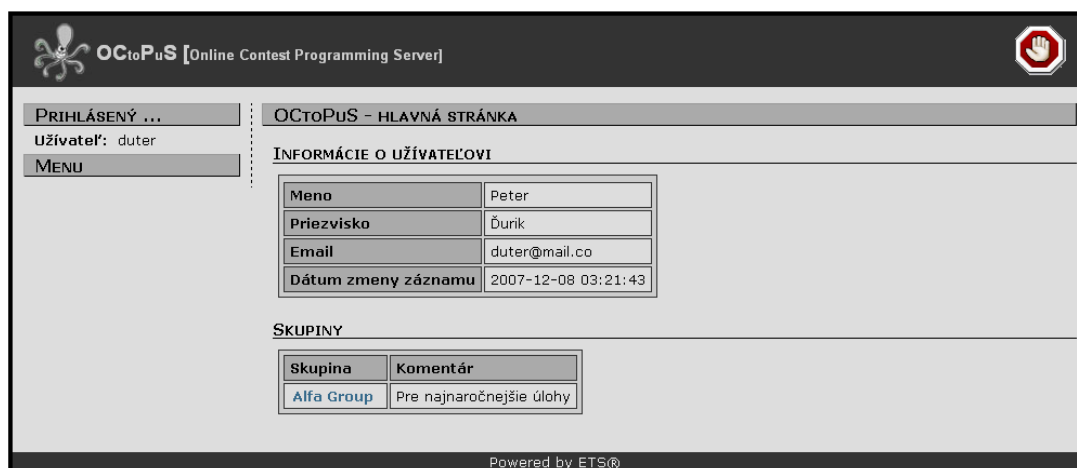
#### Prihlásenie

Vstup do aplikácie je cez ľubovoľný webový prehliadač z akéhokoľvek operačného systému. Po zadaní adresy OCPS severu sa zobrazí prihlasovacia stránka užívateľ vyplní prihlasovacie meno a heslo

**4-8: webové rozhranie - prihlásenie sa do aplikácie**

#### Hlavná stránka

Po úspešnom prihlásení sa dostaneme na hlavnú stránku. Tu sa zobrazia osobné údaje prihláseného užívateľa.



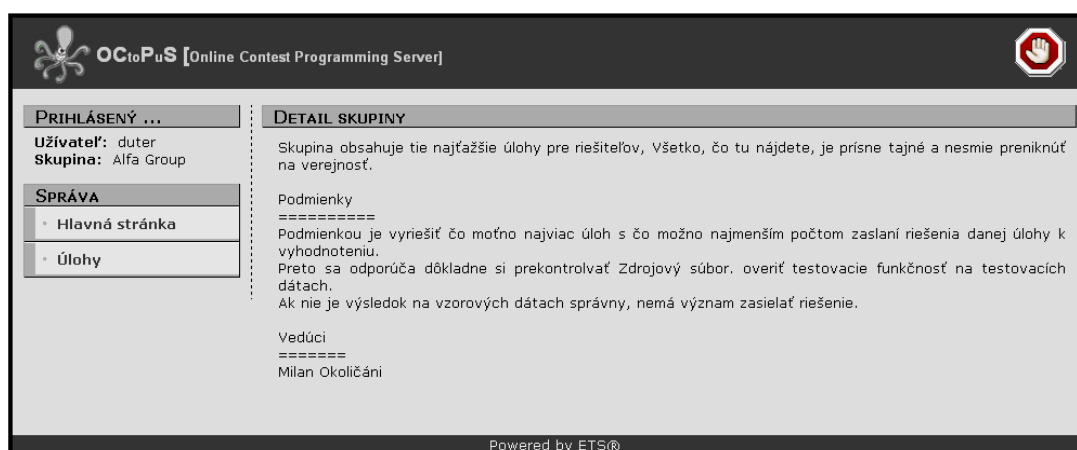
#### 4-9: webové rozhranie - hlavná stránka

Na stránke je zobrazený i zoznam skupín, do ktorých užívateľ patrí spolu so stručným komentárom. Pre vstup do ľubovoľnej skupiny postačí kliknúť na názov skupiny ľavým tlačidlom myši.

### Detail skupiny

Do tejto sekcie sa dostaneme zo *hlavnej stránky*. Nájdeme tu detailnejší popis skupiny. Je možné tu nájsť rôzne údaje prislúchajúce k danej skupine, pravidlá a podobne. Záleží na správcovi, aké informácie sa tu objavujú.

Z tohto bodu je možné sa cez menu dostať na *hlavnú stránku* alebo získať zoznam úloh.



#### 4-10: webové rozhranie - detail skupiny

## Zoznam úloh

Nájdeme tu zoznam úloh pre konkrétnu skupinu v prehľadnej tabuľke. Sú tu zobrazené všetky úlohy, ktoré prebehli, alebo sú aktívne spolu s doplňujúcimi informáciami.



Tabuľka obsahuje:

- *Aktuálne úlohy* – úlohy, ktorých riešenia je možné v danom čase zasielať
- *Neaktívne úlohy* – úlohy, ktorých čas ukončenia už uplynul.

Časti tabuľky:

- *Úloha* – názov úlohy
- *Zahájenie* – čas, od ktorého je možné zasielať riešenia úlohy
- *Koniec* – čas, do kedy bude možné zasielať riešenia úlohy
- *Zadanie úlohy* – po kliknutí na tlačidlo sa zobrazí zadanie úlohy
- *Zaslať riešenie* - kliknutie na tlačidlo umožní zaslať riešenie úlohy
- *Vyhodnotenie úlohy* – po kliknutí na tlačidlo sa zobrazí vyhodnotenie zaslaného riešenia úlohy
- *Zoznam zaslaných riešení* – po kliknutí na tlačidlo sa zobrazí vyhodnotenie všetkých doteraz zaslaných riešení úlohy

Cez menu je možné sa dostať na *hlavnú stránku* alebo na *detail skupiny*.


**OCtoPuS** [Online Contest Programming Server]
 

**PRIHLÁSENÝ ...**  
 Užívateľ: duter  
 Skupina: Alfa Group

**SPRÁVA**  
 • Hlavná stránka  
 • Náhľad na skupinu

**ZOZNAM ÚLOH**

Úloha	Zahájenie	Koniec		Zadanie úlohy	Zaslať riešenie úlohy	Vyhodnotenie úlohy (posledné)	Zoznam zaslaných riešení
<b>Aktuálne úlohy</b>							
Problem B	2007-10-12 13:00:00	2008-12-04 13:00:00	✓	zadanie	zaslať	vysledok	vsetky vysledky
Problem D	2007-03-12 13:00:00	2008-12-04 13:00:00	✓	zadanie	zaslať	vysledok	vsetky vysledky
Problem E	2007-03-12 13:00:00	2008-12-04 13:00:00	✓	zadanie	zaslať	vysledok	vsetky vysledky
Problem F	2007-03-12 13:00:00	2008-08-22 13:00:00	✗	zadanie	zaslať	vysledok	vsetky vysledky
<b>Neaktívne úlohy</b>							
Problem A	2007-10-12 13:00:40	2007-12-04 13:00:00	✓	zadanie	zaslať	vysledok	vsetky vysledky

Powered by ETS®

#### 4-11: webové rozhranie - zoznam úloh



### Zadanie úlohy

Tu sa nachádza zadanie k vybranej úlohe. Dostaneme sa sem cez zo *zoznamu úloh* po kliknutí na príslušné tlačidlo.

V pravej časti hore sú odkazy rýchlej navigácie. V tomto prípade sa cez ne môžeme dostať do časti *zaslať riešenie úlohy* a *zobraziť vyhodnotenie*.

Cez menu je možné sa dostať na *hlavnú stránku*, na *detail skupiny* alebo späť na *zoznam úloh*.




**OCtoPuS** [Online Contest Programming Server]
 

**PRIHLÁSENÝ ...**  
 Užívateľ: duter  
 Skupina: Alfa Group  
 Příklad: Problem A

**SPRÁVA**  
 • Hlavná stránka  
 • Náhľad na skupinu  
 • Úlohy

**ZADANIE ÚLOHY**  
[Zaslať riešenie úlohy](#) [Vyhodnotenie úlohy \(posledné\)](#)

**ZADANIE**  

The Center City fire department collaborates with the transportation department to maintain maps of the city which reflects the current status of the city streets. On any given day, several streets are closed for repairs or construction. Firefighters need to be able to select routes from the firestations to fires that do not use closed streets.

Central City is divided into non-overlapping fire districts, each containing a single firestation. When a fire is reported, a central dispatcher alerts the firestation of the district where the fire is located and gives a list of possible routes from the firestation to the fire. You must write a program that the central dispatcher can use to generate routes from the district firestations to the fires.

[http://www.karrels.org/Ed/ACM/91/prob\\_a.html](http://www.karrels.org/Ed/ACM/91/prob_a.html)

**VSTUP**  

The city has a separate map for each fire district. Streetcorners of each map are identified by positive integers less than 21, with the firestation always on corner #1. The input file contains several test cases representing different fires in different districts. The first line of a test case consists of a single integer which is the number of the streetcorner closest to the fire. The next several lines consist of pairs of positive integers separated by blanks which are the adjacent streetcorners of open streets. (For example, if the pair 4 7 is on a line in the file, then the street between streetcorners 4 and 7 is open. There are no other streetcorners between 4 and 7 on that section of the street.) The final line of each test case consists of a pair of 0's.

**VYSTUP**  

For each test case, your output must identify the case by number (case #1, case #2, etc). It must list each route on a separate line, with the streetcorners written in the order in which they appear on the route. And it must give the total number routes from firestation to the fire. Include only routes which do not pass through any streetcorner more than once. (For obvious reasons, the fire department doesn't want its trucks driving around in circles.) Output from separate cases must appear on separate lines. The following sample input and corresponding correct output represents two test cases.

**VZOROVÝ VSTUP**  

```

6
1 2
    
```

#### 4-12: webové rozhranie - zadanie úlohy

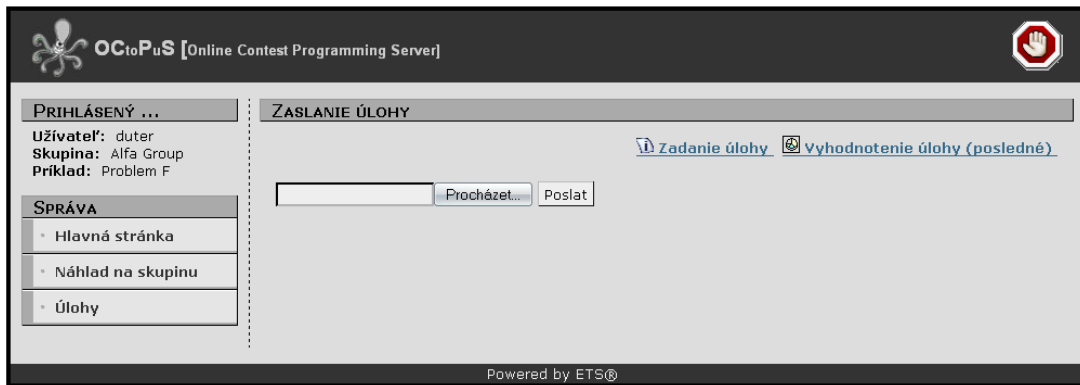
### Zaslať riešenie

Tu môžeme k danej úlohe zaslať riešenie k vyhodnoteniu. Dostaneme sa sem cez *zaslať riešenie* po kliknutí na príslušné tlačidlo. Do tejto časti sa nie je možné dostať, ak je úloha medzi *neaktívnymi úlohami* alebo úlohami úspešne vyriešenými.

To však nie je možné, ak sme už riešenie zaslali a ešte nebolo vyhodnotené. Miesto zaslania súboru budeme upozornení o danom fakte oznámením.

V pravej časti hore sú odkazy rýchlej navigácie. V tomto prípade sa cez ne môžeme dostať do časti *zaslať zadanie úlohy* a *zobraziť vyhodnotenie*.

Cez menu je možné sa dostať na *hlavnú stránku*, na *detail skupiny* alebo späť na *zoznam úloh*.



#### 4-13: webové rozhranie - zaslať riešenie úlohy

### Vyhodnotenie úlohy

Tu nájdeme výsledok vyhodnotenia posledne zaslanej úlohy, v prípade že nejaký výsledok existuje. Dostaneme sa sem cez *vyhodnotenie úlohy* po kliknutí na príslušné tlačidlo.

K videniu sú detaily:

- *čas zaslania* – čas, kedy bola úloha zaslaná riešiteľom
- *čas vyhodnotenia* – čas kedy OCPS riešenie vyhodnotil
- *pridelené body za riešenie* – pridelené body podľa výsledku vyhodnotenia
- *výsledok* – výsledok vyhodnotenia
  - *OK* – úloha bola vyriešená správne
  - *Wrong answer* – úloha prešla kompiláciou, ale výsledok na vyhodnocovacie dáta dol nesprávny
  - *Compile error* – úloha neprešla kompilátorom
  - *Time limit exceeds* – testovaná úloha sa neukončila do časového limitu.
  - *Memory limit exceeds* – testovaná úloha vyčerpala pamäťový limit.
  - *A ďalšie*
- *Výstup kompilátora* – zobrazí sa v prípade, ak je výsledok *Compile error*.

Pokiaľ úloha ešte nebola vyhodnotená, zobrazí sa o tomto mieste detailov informácia:

- Úloha ešte nebola k vyhodnoteniu zaslaná
- Úloha bola zaslaná, ale nebola vyhodnotená. Zobrazí sa čas zaslania úlohy.

V pravej časti hore sú odkazy rýchlej navigácie. V tomto prípade sa cez ne môžeme dostať do časti *zaslať zadanie úlohy* a *zaslať riešenie úlohy*.

Cez menu je možné sa dostať na *hlavnú stránku*, na *detail skupiny* alebo späť na *zoznam úloh*.

The screenshot shows the OCtoPuS web interface. At the top, there's a header with the OCtoPuS logo and name, and a 'logout' link. Below the header, the interface is divided into several sections. On the left, there's a 'PRIHLASENÝ ...' section showing user information: 'Užívateľ: duter', 'Skupina: SK-1', and 'Príklad: u1'. Below this is a 'MENU' section with links: 'HLAVNÁ STRÁNKA', 'VYBRÁŤ SKUPINU', 'NÁHLAD NA SKUPINU', and 'ÚLOHY'. The main content area is titled 'VÝSLEDOK VYHODNOCOVANIA'. It contains two buttons: 'Zadanie úlohy' and 'Zaslať riešenie úlohy'. Below these buttons is a table showing evaluation results:

Zaslané k vyhodnoteniu	2007-09-24 02:29:23
Vyhodnotené	2007-09-24 02:29:44
Pridelené body	0
Výsledok	Compile error.

Below the table is a section titled 'Výstup kompilátora' containing the following text:

```
1.c: In function 'main':
1.c:9: error: syntax error before '}' token
Exited with error status 1.
```

At the bottom of the interface, it says 'Powered by ETS®'.

#### 4-14: webové rozhranie - výsledok vyhodnotenia

### Zoznam zaslaných riešení

Tu nájdeme výsledok vyhodnotenia všetkých doteraz zaslaných riešení danej úlohy, v prípade že nejaký výsledok existuje. Dostaneme sa sem cez *vyhodnotenie úlohy* po kliknutí na príslušné tlačidlo.

Pokiaľ úloha ešte nebola vyhodnotená, zobrazí sa o tomto mieste detailov informácia:

- Úloha ešte nebola k vyhodnoteniu zaslaná

V pravej časti hore sú odkazy rýchlej navigácie. V tomto prípade sa cez ne môžeme dostať do časti *zaslať zadanie úlohy* a *zaslať riešenie úlohy*.

Cez menu je možné sa dostať na *hlavnú stránku*, na *detail skupiny* alebo späť na *zoznam úloh*.

**OCtoPuS [Online Contest Programming Server]**

**PRIHLÁSENÝ ...**  
 Uživateľ: duter  
 Skupina: Alfa Group  
 Příklad: Problem F

**SPRÁVA**  
 • Hlavná stránka  
 • Náhľad na skupinu  
 • Úlohy

**VÝSLEDOK VYHODNOCOVANIA**

[Zadanie úlohy](#) [Zaslať riešenie úlohy](#)

Zaslané k vyhodnoteniu	2007-12-09 02:21:39
Vyhodnotené	2007-12-09 02:21:44
Pridelené body	0
Výsledok	Wrong answer.

Zaslané k vyhodnoteniu	2007-12-09 02:21:16
Vyhodnotené	2007-12-09 02:21:18
Pridelené body	0
Výsledok	Caught fatal signal 11.

Zaslané k vyhodnoteniu	2007-12-09 02:20:44
Vyhodnotené	2007-12-09 02:20:47
Pridelené body	0
Výsledok	Compile error.

Zaslané k vyhodnoteniu	2007-12-09 02:20:15
Vyhodnotené	2007-12-09 02:20:16
Pridelené body	0
Výsledok	Wrong answer.

**Výstup kompilátora**

```
18.c: In function `main':
18.c:39: error: syntax error at end of input
Exited with error status 1.
```

Powered by ETS®

#### 4-15: webové rozhranie - zoznam zaslaných riešení

## Odhlásenie

Z aplikácie je možné sa odhlásiť po kliknutí ľavým tlačidlom myši na symbol ryky v červenom osemuholníku.



#### 4-16: webové rozhranie - odhlásenie sa

### 4.4.2 OCPS manager

*OCPS manager* je časť, ktorá sa stará o zbieranie zaslaných riešení riešiteľmi, o ich pridelovaní vyhodnocovaciemu severu *JUDGE* a následnú aktualizáciu dát v databáze požiadaviek a výsledkov vyhodnocovania.

*OCPS manager* je nainštalovaný na rovnakom stroji ako *webový server OCPS*. Inštalácia bola prevedená v kapitole 0 v časti *OCPS Server*, v krokoch 5 a 6. Pokiaľ zostala zachovaná pôvodná adresárová štruktúra, nájdeme ho v */aux/ocps*.

Premiestnením sa do tohto adresára spustíme *manager* ako užívateľ *ocps* príkazom:

```
# ./ocps_manager
```

Server zastavíme zaslaním *kill* signálu stlačením kombinácie kláves *Ctrl+c*.

## 5 ZÁVER

OCtoPuS vznikol ako aplikácia pre vyhodnocovanie úloh z programovania. Pôvodný zámer bol napísať aplikáciu, ktorá bude podobná existujúcim univerzitným *Online Judge*. Nakoniec som sa rozhodol vypracovať aplikáciu, ktorá bude môcť byť používaná v školách ako prostriedok k vyhodnocovaniu napríklad domácich úloh z programovania.

Pri práci som sa stretol z rôznymi problémami. V začiatku to bola pre mňa predovšetkým úplne nová téma. Samotné vnorenie sa do problematiky a pochopenie, ako podobné systémy fungujú stálo istú námahu. Postupne vznikali nápady a aplikácia dostávala reálnu podobu. I keď sa môže zdať, že je systém založený na webovom rozhraní, hodne práce bolo venované prepojeniu jednotných častí, pretože celý systém je poskladaný zo samostatných komponentov. Nakoniec ale jedným z hlavných cieľov bolo vytvorenie aplikácie, ktorá bude ako pre užívateľa tak i pre administrátora užívateľsky príjemná.

Systém nie je úplne dokonalý, ale je funkčný. Cieľom nebolo napísať aplikáciu, ktorá bude schopná odolávať obrovskému náporu a záťaži zo strany užívateľov, ale bol tu zámer zistiť ako taký online evaluátor funguje a pokúsiť sa zostaviť funkčný prototyp.

Myslím, že sa podarilo navrhnuť a zostaviť systém, ktorý je v praxi uplatniteľný. Rovnako dobre sa dá použiť ako server pre korešpondenčné semináre, tak i aplikácia pre výučbu a tréning si úloh v programovaní. Mnoho ľudí, ktorí programovať len začínajú by takúto možnosť uvítali. A rovnako tak je vhodný na použitie pri výučbe na školách ako pomôcka učiteľa. Automatizuje prácu a zaručuje objektívne vyhodnocovanie úloh. Stále tu ale ostáva možnosť vyhodnotiť úlohy ručne.

Ako najväčšie klady návrhu systému vidím:

- jeho univerzálnosť v použití
- distribuovaný systém vyhodnocovania, kde sa na vyhodnocovaní môže súčasne podieľať viac evaluátorov
- príjemné užívateľské rozhranie.

Aplikácia sa dá v mnohých smeroch rozširovať a zdokonaľovať aby priniesla komplexné uspokojenie vo všetkých smeroch.

Je tu možnosť rozšíriť aplikáciu o:

- zobrazovanie štatistík vyhodnocovaných úloh. Tu sa jedná o získanie a spravovanie údajov z tabuľky výsledkov a následné zapracovanie do prezenčnej vrstvy.
- prístup administrátora k zaslaným súborom so zdrojovými kódmi, či vytváranie konfiguračných súborov pre evaluátor priamo z webového rozhrania.
- možnosť vkladať do zadania problémov obrázky. Zadanie by tým získalo na názornosti. Vo väčšine online validátoroch je zadanie úloh riešené formou textového popisu. Tak je to i v prípade OCPS.

## 6 ZDROJE

- [1] UVa Online Judge [online]  
*<http://icpcres.ecs.baylor.edu/onlinejudge/index.php>*
- [2] El Judge - Online MIPT programming contest [online]  
*<http://acm.mipt.ru/judge>*
- [3] FZU Online Judge - Fuzhou University [online]  
*<http://acm.fzu.edu.cn>*
- [4] ZHEJIANG niversity Online Judge [online]  
*<http://acm.zju.edu.cn>*
- [5] TJU Online Judge – Tiabjin University [online]  
*<http://acm.tju.edu.cn/toj>*
- [6] Timus Online Judge - Ural State University [online]  
*<http://acm.timus.ru>*
- [7] Sphere Online Judge [online]  
*<https://www.spoj.pl>*
- [8] Online Contester – Satarov State University [online]  
*<http://acm.sgu.ru>*
- [9] SHAHRIAR, Manzoor: Analyzing Programming Contest Statistics [online]  
*[http://online-judge.uva.es/problemset/13\\_Manzoor\\_rev.pdf](http://online-judge.uva.es/problemset/13_Manzoor_rev.pdf)*
- [10] MO Eval - The MO Contest Environment [online]  
*<http://mj.ucw.cz/mo-eval>*